

Universidad Politécnica del Estado de Morelos



"HealthyView: Sistema Web de Monitoreo Nutricional y Hábitos Saludables
para el impulso del Bienestar y Rendimiento Académico"

ESTANCIA II

INGENIERÍA EN TECNOLOGÍAS DE LA INFORMACIÓN

P r e s e n t a

Ramírez Pérez Mauricio - RPMO230436

Ginera López David - GLDO230445

7°B

Profesora: Sandra Elizabeth León Sosa

JIUTEPEC, VIERNES 19 DE SEPTIEMBRE DEL 2025.

Índice

Índice	2
Índice de figuras	5
Índice de Tablas.....	7
Capítulo I - Análisis empresarial	9
Introducción	9
1.1 Datos de la empresa	9
- Nombre de la empresa	9
- Dirección	9
- Sector empresarial	9
- Giro empresarial	9
- Tamaño	9
- Historia	10
- Misión	12
- Visión	13
- Valores	13
- Política de calidad	13
- Política ambiental	14
- Política de igual laboral y no discriminación	14
- Modelo organizacional	15
- Análisis FODA	15
1.2 Antecedentes	17
1.3 Definición del problema	18
1.4 Objetivo General y específicos	18
1.5 Justificación	18
1.6 Alcances y limitaciones	19
- Alcances	19
- Limitaciones	19
1.7 Entregables.....	19
1.8 Calendario de actividades	20
- Fase 1. Inicio (16 – 21 de septiembre)	20
- Fase 2. Sprint 1 (22 de septiembre – 05 de octubre)	21

- Fase 3. Sprint 2 (06 – 19 de octubre)	21
- Fase 4. Sprint 3 (20 de octubre – 02 de noviembre)	21
- Fase 5. Sprint 4 (03 – 09 de noviembre).....	21
- Fase 6. Cierre (10 – 15 de noviembre)	22
Capítulo II - Modelado de Negocio.....	23
2.1 Desarrollo	23
- Técnica de Obtención de requisitos	23
- Evidencia de la Técnica de requisitos.....	23
- Metodología	24
- Especificación de requisitos	25
Capítulo III - Modelado del Sistema Administración / Investigación	33
3.1 Diseño de requisitos funcionales.	33
3.2 Arquitectura del sistema.	40
3.3 Diseño de la interfaz de usuario.....	41
3.4 Diseño de base de datos.	52
- Modelo Relacional	52
- Diccionario de datos	52
- Prueba de la base de datos	56
Capitulo IV - Investigación de herramientas y tecnologías para el desarrollo el proyecto de TI.....	63
4.1 Tecnologías investigadas.....	63
Lenguajes de programación.....	63
Entornos de Desarrollo.....	65
Tecnologías de Presentación.....	67
Base de Datos	69
Servidor y Herramientas de Integración	71
Servidor Web	73
4.2 Selección de tecnologías	75
Lenguaje de programación	75
Entorno de desarrollo	75
Lenguajes de presentación	76
Base de datos.....	76
Servidor local	77
Servidor web.....	77

Conclusión	78
Bibliografía	79

Índice de figuras

Ilustración 1: Organigrama del área de ITI	15
Ilustración 2 - Imagen Cronograma (Gantt)	22
Ilustración 3 Diagrama de actividad RF-01: Inicio de sesión	33
Ilustración 4 Diagrama de actividad RF-02: Gestión de pacientes	34
Ilustración 5 Diagrama de actividad RF-03: Gestión de Médicos.....	34
Ilustración 6 Diagrama de actividad RF-04: Gestión de administradores	35
Ilustración 7 Diagrama de actividad RF-05: Gestión de perfiles de participantes	35
Ilustración 8 Diagrama de actividad RF-06: Gestión de recetas médicas	36
Ilustración 9 Diagrama de actividad RF-07: Gestión de actividades saludables	36
Ilustración 10 Diagrama de actividad RF-08: Gestión de citas medicas.....	37
Ilustración 11 Diagrama de actividad RF-09: Asignación de recetas a pacientes	37
Ilustración 12 Diagrama de actividad RF-10: Registro de progreso de actividades.....	38
Ilustración 13 Diagrama de actividad RF-11: Registro de seguimientos clínicos	38
Ilustración 14 Diagrama de actividad RF-12: Generación de reportes	39
Ilustración 15 Diagrama de actividad RF-13: Publicación de mensajes en foro	39
Ilustración 16 Diagrama de actividad RF-14: Respaldo y restauración de la base de datos	40
Ilustración 17 - Diagrama Modelo-Vista-Controlador (MVC).....	40
Ilustración 18 Boceto del inicio de sesión	42
Ilustración 19 boceto del registro de nuevo paciente	43
Ilustración 20 Boceto del dashboard del paciente	44
Ilustración 21 Boceto de la gestión de actividades saludables	45
Ilustración 22 Boceto del monitoreo del medico	46
Ilustración 23 Boceto de la gestión de citas	47
Ilustración 24 Boceto de la gestión de recetas	48
Ilustración 25 Boceto del panel del administrador	49
Ilustración 26 Boceto de la vista de impresión de los reportes	50
Ilustración 27 Boceto de la vista desde el doctor.....	51

Ilustración 28 Modelo Relacional de nuestras tablas que conforman la base de datos (BD)	52
Ilustración 29 Conexión a base BD	56
Ilustración 30 Tabla paciente con registros exitosos	57
Ilustración 31 Tabla medico con registros exitosos	57
Ilustración 32 Tabla administrador con registros exitosos	58
Ilustración 33 Tabla sesion con registros exitosos	58
Ilustración 34 Tabla receta con registros exitosos	59
Ilustración 35 Tabla recetaltem con registros exitosos.....	59
Ilustración 36 Tabla actividad con registros exitosos.....	60
Ilustración 37 Tabla actividadPaciente con registros exitosos.....	60
Ilustración 38 Tabla cita con registros exitosos.....	61
Ilustración 39 Tabla seguimiento con registros exitosos	61
Ilustración 40 Tabla foro con registros exitosos.....	62
Ilustración 41 Tabla reporte con registros exitosos	62

Índice de Tablas

Tabla 1 Cronograma de Actividades para proyecto	20
Tabla 2 RFN1 - Inicio de sesión	25
Tabla 3 RFN2 - Gestión de pacientes	25
Tabla 4 RFN3 - Gestión de médicos	26
Tabla 5 RFN4 - Gestión de administradores	26
Tabla 6 RFN5 - Gestión de perfiles	26
Tabla 7 RFN6 - Gestión de recetas médicas	27
Tabla 8 RFN7 - Gestión de actividades saludables	27
Tabla 9 RFN8 - Gestión de citas médicas	27
Tabla 10 RFN9 - Asignación de recetas a pacientes	28
Tabla 11 RFN10 - Registro de progreso de actividades	28
Tabla 12 RFN11 - Registro de seguimientos clínicos	28
Tabla 13 RFN12 - Generación de reportes	29
Tabla 14 RFN13 - Publicación de mensajes en foro	29
Tabla 15 RFN14 - Respaldo y restauración de la base de datos	29
Tabla 16 RFF1 - Mensaje de error	30
Tabla 17 RFF2 - Error de inicio de sesión	30
Tabla 18 RFF3 - Validación de campos	30
Tabla 19 RIN1 - Usabilidad	31
Tabla 20 RCA1 - Diseño responsivo	31
Tabla 21 REV1 - MVC (Modelo vista-controlador)	31
Tabla 22 RPR1 - Tiempo de desarrollo	32
Tabla 23 RSO1 - Conexión a internet	32
Tabla 24 RSO2 – Navegador compatible	32
Tabla 25 Diccionario de datos de la tabla actividad	52
Tabla 26 Diccionario de datos de la tabla actividadPaciente	53
Tabla 27 Diccionario de datos de la tabla administrador	53

Tabla 28 Diccionario de datos de la tabla cita	53
Tabla 29 Diccionario de datos de la tabla foro	54
Tabla 30 Diccionario de datos de la tabla medico.....	54
Tabla 31 Diccionario de datos de la tabla paciente.....	54
Tabla 32 Diccionario de datos de la tabla receta	55
Tabla 33 Diccionario de datos de la tabla recetaltem	55
Tabla 34 Diccionario de datos de la tabla reporte	55
Tabla 35 Diccionario de datos de la tabla seguimiento.....	56
Tabla 36 Diccionario de datos de la tabla sesion	56

Introducción

En la actualidad, las universidades enfrentan el desafío de promover estilos de vida saludables entre sus estudiantes, pues el bienestar físico y mental influye directamente en el rendimiento académico y en el desarrollo personal. Sin embargo, gran parte de la población universitaria presenta hábitos nutricionales poco adecuados y carece de herramientas que les permitan dar un seguimiento integral a su estado de salud.

Ante esta situación, surge la necesidad de implementar soluciones tecnológicas que integren la nutrición, la actividad física, el manejo del estrés y el desempeño escolar en un mismo espacio. Con esta finalidad, se plantea el desarrollo del sistema web HealthyView, cuyo propósito es centralizar el monitoreo de la salud y los hábitos de los estudiantes universitarios.

1.1 Datos de la empresa

- Nombre de la empresa

Universidad Politécnica del Estado de Morelos

- Dirección

Boulevard Cuauhnáhuac #566, Col. Lomas del Texcal, Jiutepec, Morelos. CP 62550

- Sector empresarial

Sector TERCIARIO (turismo, educación, salubridad, finanzas y seguros).

- Giro empresarial

Público

- Tamaño

Grande (Más de 251 empleados)

- Historia

En el 2003 se dio inicio a un estudio de viabilidad para la creación de la Universidad Politécnica del Estado de Morelos, liderados por la Secretaría de Educación del Gobierno del Estado. En el estudio para determinar la instalación de la Universidad Politécnica en el estado de Morelos, se tomaron en consideración los distintos proyectos estratégicos federales y estatales con que se cuenta respecto a la educación de nivel superior, así como los aspectos geográficos y demográficos del Estado dentro del entorno nacional.

Dentro de los proyectos estratégicos de educación y de desarrollo económico de los gobiernos Estatal y Federal que apoyan la instalación, operación y desarrollo de una Universidad Politécnica se encuentran: El Programa Nacional de Educación 2001-2006, el Programa Nacional de Desarrollo de la Educación Tecnológica 2001-2006, el Plan Estatal de Desarrollo del Estado de Morelos, el Programa Educativo Estatal Morelos 2001-2006, el Programa Estatal de Desarrollo Urbano 2001-2006 y el Proyecto Gran Visión Morelos 2025.

¿Por qué Jiutepec?

Así como el estado de Morelos se ubica en una situación privilegiada por su cercanía a la principal concentración urbana del país; el municipio de Jiutepec se ubica en un área de oportunidad por su colindancia a la capital del Estado y su cercanía a los principales desarrollos urbanos, comerciales e industriales. De igual forma, la ubicación de la Universidad Politécnica del Estado de Morelos en el municipio de Jiutepec es estratégica por localizarse en la zona potencial para uso urbano, por encontrarse en este municipio la zona industrial más grande e importante del Estado y por no contar con una institución de educación superior.

Jiutepec es uno de los 33 municipios que conforman el estado de Morelos, localizándose al norte del Estado. Se ubica geográficamente entre los paralelos 18° 53' de latitud norte y los 99° 10' de longitud oeste del meridiano de Greenwich, a una altura de 1,350 metros sobre el nivel del mar. Tiene una superficie de 70.45 km², cifra que representa el 1.42% de la superficie total del Estado. El terreno que

ocupa el municipio en su mayoría es plano, sus áreas accidentadas son parte de la sierra de tetillas y los cerros donde se encuentran las colonias Vista Hermosa, la Independencia y Cerro de la Corona.

Jiutepec se encuentra comunicado por la carretera federal Cuernavaca-Cuautla, la carretera estatal Tejalpa-Zacatepec y el boulevard Cuauhnáhuac que constituye uno de los principales accesos a Cuernavaca; cuenta además con la carretera Jiutepec–Atlacomulco-Cuernavaca y Jiutepec-Palmira. La autopista Siglo XXI permite que el estado de Puebla quede a menos de una hora de la ubicación de la Universidad Politécnica del Estado de Morelos, lo que se traduce como otra ventaja al facilitar a los jóvenes de esas latitudes trasladarse a cursar estudios superiores en el Estado.

Limita al norte con Tepoztlán y Cuernavaca, al oriente con Yautepec, al sur con Emiliano Zapata y al poniente con Cuernavaca y Temixco, considerando la población de Jiutepec y sus municipios vecinos, concentran a 573,578 habitantes, que representan el 37% de la población total del Estado.

Después de Cuernavaca, el municipio de Jiutepec es el que registro el mayor crecimiento poblacional en el período 1990-2000, así mismo la densidad poblacional en este municipio es la más alta del Estado con 3,464 hab/Km².

Las investigaciones realizadas arrojaron datos que percibían un periodo de expansión de la demanda social de la educación superior en nuestro Estado. El índice de cobertura de nivel licenciatura en el estado de Morelos se mantuvo rezagado respecto a la media nacional; tan sólo en el ciclo 2002-2003, Morelos atendió al 16.8% de los jóvenes en edades comprendidas entre los 19 y 23 años, mientras que la media nacional fue del 19.1%. Sin lugar a dudas, la demanda para realizar estudios de nivel superior se incrementará consistentemente en los próximos 10 años, lo que hizo especialmente oportuna y necesaria la creación de una Universidad pública. Así mismo, se pronosticó que un lapso de 12 años entre 2004 y 2016 era suficiente para consolidar la oferta educativa de la nueva universidad, lo que será fundamental para asegurar que ésta se mantenga como una opción atractiva. El 7 de julio de 2004, la Universidad Politécnica del Estado de

Morelos, UPEMOR, fue fundada bajo el decreto número doscientos ochenta y ocho, publicado en el Periódico Oficial “Tierra y Libertad”, órgano del Gobierno del Estado Libre y Soberano de Morelos. Su Rector fundador fue el M. A. Iván Alberto Elizondo Cortina, como Secretario Académico el Dr. Jaime Padilla Acero y como Secretario Administrativo el C. P. José Villegas Vázquez.

Los principales objetivos de la Universidad son:

- I. Impartir educación superior en los niveles de licenciatura, especialización tecnológica y otros estudios de postgrado, así como cursos de actualización en sus diversas modalidades, para preparar profesionales con una sólida formación técnica y en valores, conscientes del contexto estatal y nacional en lo económico, político y social;
- II. Llevar a cabo investigación aplicada y desarrollo tecnológico, pertinentes para el desarrollo económico y social,
- III. Difundir el conocimiento y la cultura a través de la extensión universitaria;
- IV. Prestar servicios tecnológicos y de asesoría, que contribuyan a mejorar el desempeño de las empresas y otras organizaciones en la región y del Estado principalmente, y
- V. Impartir programas de educación continua con orientación a la capacitación para el trabajo y al fomento de la cultura tecnológica en la región y en el estado.

- Misión

Somos una Universidad pública de calidad dedicada a la formación integral de estudiantes, a la innovación e investigación científica y tecnológica para atender las necesidades del sector productivo y de la sociedad en general, comprometida como institución con el crecimiento sostenible dentro del marco regulatorio aplicable.

- Visión

Ser una Universidad reconocida por su liderazgo en el Subsistema de Universidades Tecnológicas, comprometida con la calidad educativa, innovación, ciencia, tecnología y medio ambiente, contribuyendo en la construcción de la prosperidad compartida de la región y el país.

- Valores

- I. Cooperación
- II. Cuidado del Entorno Cultural y Ecológico
- III. Equidad de género
- IV. Igualdad
- V. No Discriminación
- VI. Interés Público
- VII. Liderazgo
- VIII. Respeto
- IX. Respeto a los Derechos Humanos

- Política de calidad

Somos una Universidad pública comprometida con la prosperidad compartida y la calidad educativa para la formación integral de estudiantes mediante la creación, aplicación y difusión del conocimiento, la innovación e investigación científica y tecnológica, con una visión de mejora continua; orientada a la productividad, atendiendo las demandas del sector productivo y la sociedad en general, asegurando el cumplimiento de la normatividad y los requisitos aplicables, siendo un referente para el crecimiento sostenible del estado de Morelos.

- Política ambiental

"Somos una universidad pública comprometida con la mitigación y prevención de la contaminación, la responsabilidad en el uso de los recursos naturales y la protección del ambiente, afrontamos riesgos con una visión de sustentabilidad y de mejora continua, asegurando el cumplimiento de la legislación aplicable y otros requisitos, comprometidos con el bien común y como un referente en el estado de Morelos"

- Política de igual laboral y no discriminación

Somos una institución pública de educación superior comprometida en promover la igualdad en el ámbito laboral y educativo, brindando las mismas oportunidades para hombres y mujeres en el acceso a la educación y al empleo generando un ambiente sano y condiciones armoniosas, a través de acciones a favor de nuestra comunidad.

Prohibimos la discriminación (1) y queda estrictamente prohibida cualquier forma de maltrato, violencia y segregación de las autoridades de la Upemor hacia el personal y entre el personal en materia de:

- o Apariencia física
- o Cultura
- o Discapacidad
- o Idioma
- o Sexo
- o Género
- o Preferencias sexuales
- o Edad
- o Condición social, económica, de salud o jurídica
- o Embarazo
- o Estado civil o conyugal

- ### - Modelo organizacional

[illegible]

Ilustración 1: Organigrama del área de ITI

- Análisis FODA

- Conocimientos en programación, bases de datos, redes y administración de sistemas adquiridos en la carrera.

- Capacidad para adaptarse a diferentes lenguajes y herramientas tecnológicas.
- Acceso a recursos universitarios y asesoría docente.
- Habilidades para el trabajo en equipo y la resolución de problemas.

Oportunidades

- Creciente demanda de soluciones tecnológicas en todos los sectores empresariales.
- Posibilidad de aplicar innovaciones como inteligencia artificial, big data y seguridad informática.
- Espacios para realizar prácticas profesionales y vinculación con empresas locales.
- Acceso a certificaciones y capacitaciones en tecnologías actuales (Cisco, bases de datos, cloud, etc.).

Debilidades

- Experiencia práctica limitada en entornos empresariales reales.
- Dependencia de recursos de la universidad (laboratorios, software con licencia).
- Falta de especialización en un área concreta de TI.
- En ocasiones, dificultad para gestionar el tiempo entre estudios y proyectos externos.

Amenazas

- Alta competencia con egresados de otras universidades y profesionales con más experiencia.
- Rápido avance tecnológico que vuelve obsoletos ciertos conocimientos.
- Brechas de recursos tecnológicos en algunas organizaciones (infraestructura limitada).
- Posibles recortes o falta de inversión en tecnología en empresas locales.

1.2 Antecedentes

La relación entre la nutrición, los hábitos alimentarios y el desempeño académico de los estudiantes universitarios ha sido un tema de creciente interés en México. Sin embargo, la evidencia disponible sugiere que aún existen limitaciones significativas en los conocimientos nutricionales de los propios estudiantes, incluso en aquellos que cursan programas relacionados con la salud.

Un estudio realizado en Chiapas por López-Gutiérrez, Castañeda-Limón, López-Aguilar y López-García (2017) evaluó los conocimientos nutricionales de estudiantes universitarios de nutriología, medicina y enfermería mediante un cuestionario estructurado. Los resultados revelaron que, aunque los alumnos de nutriología obtuvieron los puntajes más altos, ninguno de los grupos alcanzó un nivel de suficiencia considerado aceptable ($\geq 80\%$). En particular, los estudiantes de enfermería presentaron las puntuaciones más bajas en temas clave como grupos de alimentos, nutrientes, enfermedades relacionadas con la alimentación e higiene alimentaria. Estos hallazgos evidencian la necesidad de reforzar la enseñanza de nutrición en los planes de estudio de carreras del área de la salud, dado que los futuros profesionales se convierten en agentes esenciales para la promoción de hábitos saludables en la población general.

Por su parte, Gil-Romo (2007) realizó una revisión de estudios sobre alimentación y nutrición en México con un enfoque de género. En este trabajo se identificó que, si bien existe un cúmulo de investigaciones sobre estado nutricional, hábitos alimentarios y desigualdades socioeconómicas, aún son escasos los estudios que vinculen directamente la nutrición con el rendimiento académico. Además, se resaltó que el género constituye un factor importante que influye en la manera en que hombres y mujeres adoptan hábitos alimentarios, lo cual debe considerarse al diseñar estrategias de intervención o monitoreo.

Ambos trabajos ponen en evidencia vacíos en la literatura nacional: por un lado, el limitado nivel de conocimientos nutricionales de los propios estudiantes universitarios de carreras de salud; y, por otro, la falta de investigaciones que unan de manera explícita la nutrición y los hábitos saludables con indicadores académicos. En este sentido, el desarrollo de un sistema web como HealthyView resulta pertinente, ya que permitiría monitorear de manera sistemática los hábitos alimentarios, brindar retroalimentación personalizada y generar evidencia longitudinal sobre la relación entre el bienestar nutricional y el rendimiento académico en estudiantes universitarios.

1.3 Definición del problema

Las universidades mexicanas enfrentan el reto de promover estilos de vida saludables entre sus estudiantes, quienes se encuentran en una etapa de formación crítica tanto profesional como personal. Sin embargo, actualmente no existen sistemas institucionales que integren en un solo espacio la recopilación, análisis y seguimiento de información sobre hábitos alimentarios, actividad física, estrés académico y rendimiento escolar. Como consecuencia, los programas de promoción de la salud se realizan de manera aislada, sin datos longitudinales ni evidencia sólida de su impacto. Esta carencia de herramientas de monitoreo integral constituye un problema que limita la capacidad de las instituciones educativas para diseñar y evaluar estrategias efectivas en favor del bienestar y rendimiento académico de los universitarios.

1.4 Objetivo General y específicos

Objetivo General:

Desarrollar un sistema web *HealthyView* que permita el monitoreo integral de la nutrición, los hábitos saludables de los estudiantes universitarios.

Objetivos Específicos:

- Implementar un sistema de gestión de usuarios con roles diferenciados (administrador, médico y paciente), garantizando autenticación y seguridad en el acceso.
- Desarrollar un módulo para que los médicos registren, gestionen y asignen recetas a pacientes en función de su estado de salud.
- Desarrollar un módulo de gestión de actividades que permita a los médicos recomendar prácticas saludables y a los pacientes consultarlas y dar seguimiento.

1.5 Justificación

El sistema web *HealthyView* busca centralizar el monitoreo de la nutrición y los hábitos saludables de los estudiantes universitarios, ofreciendo módulos de gestión de recetas, actividades, citas y seguimiento médico. Además, proporcionará reportes estadísticos y un foro motivacional que fomentará la participación y el compromiso estudiantil. De esta forma, la plataforma no solo facilitará la gestión médica y administrativa, sino que también generará evidencia útil para relacionar el bienestar nutricional con el rendimiento académico, apoyando la toma de decisiones institucionales en materia de salud universitaria.

1.6 Alcances y limitaciones

- Alcances

- Se establecerá un **sistema de gestión de citas** que permitirá a los pacientes y doctores programar, modificar y controlar consultas de manera eficiente.
- El sistema contará con un **módulo de seguimiento de bienestar**, en el que se registran indicadores de salud de los pacientes y se visualizará su evolución mediante tablas históricas y reportes personalizados.
- Se implementará un **foro motivacional** que fomentará la interacción entre pacientes, compartiendo experiencias y mensajes de apoyo para impulsar la motivación comunitaria.
- El sistema generará **reportes estadísticos clave** (número y porcentaje de pacientes por género, citas por doctor, actividades asignadas, ranking de participación), útiles para la toma de decisiones y la evaluación institucional.
- El administrador podrá controlar permisos, gestionar usuarios y acceder a reportes globales para supervisar el funcionamiento del sistema.

- Limitaciones

- El sistema está dirigido principalmente a **estudiantes universitarios**, limitando su alcance a un sector específico de la población.
- No sustituye una consulta médica presencial, funciona únicamente como una herramienta de **apoyo y seguimiento**.
- Requiere conexión estable a internet y el uso de un navegador actualizado para acceder al sistema.

1.7 Entregables

- Código fuente
- Base de datos
- Sistema Web

1.8 Calendario de actividades

Fase	Fechas	Actividades principales	Entregable
Inicio	16 – 21 sep	Alcance, backlog (lista de tareas), diseño preliminar	Documento de planificación
Sprint 1	22 sep – 05 oct	BD, login, gestión de usuarios	Módulo de autenticación (login funcional)
Sprint 2	06 – 19 oct	Recetas (doctor), actividades (paciente)	Flujo médico–paciente
Sprint 3	20 oct – 02 nov	Citas (doctor), seguimiento (paciente-doctor)	Módulos administrativos
Sprint 4	03 – 09 nov	Foro (en sistema web) y reportes (BD)	Interacción + estadísticas
Cierre	10 – 15 nov	Pruebas, documentación, entrega final	Sistema de <i>HealthyView</i> listo

Tabla 1 Cronograma de Actividades para proyecto

El proyecto se ejecutará en el periodo comprendido entre el **16 de septiembre de 2025 y el 15 de noviembre de 2025**, distribuyéndose en cinco fases principales: inicio, cuatro sprints de desarrollo y una fase de cierre.

- Fase 1. Inicio (16 – 21 de septiembre)

- Definición del alcance y objetivos del proyecto.
- Elaboración del **Product Backlog** (lista priorizada de requisitos).
- Diseño preliminar de la arquitectura del sistema y la base de datos.

- **Entregable:** Documento de planificación y backlog inicial.

- Fase 2. Sprint 1 (22 de septiembre – 05 de octubre)

- Implementación de la base de datos inicial.
- Desarrollo del módulo de gestión de usuarios (roles: administrador, médico y paciente).
- Configuración del sistema de inicio de sesión.
- **Entregable:** Sistema con login funcional y base de datos conectada.

- Fase 3. Sprint 2 (06 – 19 de octubre)

- Desarrollo del módulo de recetas (registro, edición y asignación por parte del médico).
- Desarrollo del módulo de actividades (registro de recomendaciones y consulta por parte del paciente).
- **Entregable:** Flujo funcional: *“médico asigna receta/actividad → paciente la visualiza”*.

- Fase 4. Sprint 3 (20 de octubre – 02 de noviembre)

- Implementación de la gestión de citas (programación, modificación y control).
- Creación de la tabla de seguimiento (registro histórico de indicadores de bienestar de cada paciente).
- **Entregable:** Módulo de citas y seguimiento integrado al sistema.

- Fase 5. Sprint 4 (03 – 09 de noviembre)

- Implementación del foro motivacional (publicaciones y comentarios de pacientes).
- Desarrollo de reportes estadísticos:

- Número y porcentaje de pacientes por género.
- Número y porcentaje de citas por doctor.
- Número y porcentaje de actividades asignadas.
- Ranking de participación de pacientes.
- **Entregable:** Foro activo y módulo de reportes.

- Fase 6. Cierre (10 – 15 de noviembre)

- Integración final de todos los módulos desarrollados.
- Pruebas integrales y corrección de errores.
- Elaboración de documentación técnica y manual de usuario.
- Despliegue en servidor o entorno local de prueba.
- **Entregable final:** Sistema web *HealthyView* funcional con documentación y evidencias para su presentación.

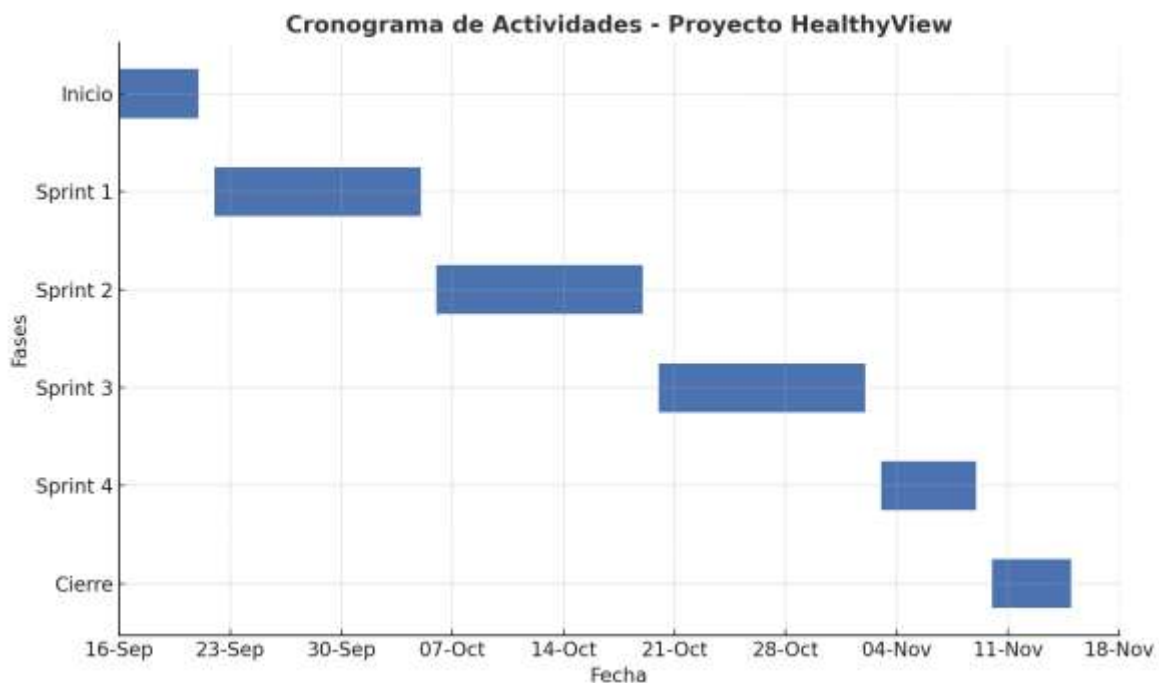


Ilustración 2 - Imagen Cronograma (Gantt)

2.1 Desarrollo

- Técnica de Obtención de requisitos

Para la obtención de los requisitos del proyecto HealthyView: Sistema Web de Monitoreo Nutricional y Hábitos Saludables, se decidió emplear la técnica de encuesta. Esta técnica fue seleccionada debido a que permite recopilar de manera estructurada y directa las necesidades y expectativas del cliente o dueño del proyecto respecto al sistema a desarrollar.

La aplicación de la encuesta se realizó de manera simulada durante la semana del 22 al 26 de septiembre de 2025, en un entorno académico de la Universidad Politécnica del Estado de Morelos, teniendo como participante al supuesto representante institucional del área de servicios estudiantiles, considerado como el dueño del proyecto.

Con esta técnica fue posible obtener una visión clara de los requerimientos funcionales y no funcionales del sistema, lo que constituye la base para el diseño y desarrollo de la solución tecnológica.

- Evidencia de la Técnica de requisitos

Para recolectar los requisitos del sistema HealthyView se diseñó un cuestionario aplicado al supuesto dueño del proyecto. El objetivo de estas preguntas fue identificar las necesidades principales, los módulos indispensables y las expectativas en cuanto a funcionalidades, seguridad y usabilidad del sistema.

Las siguientes preguntas fueron consideradas las más relevantes para guiar el levantamiento de requisitos:

1. ¿Cuál es el principal objetivo que espera cumplir con la implementación del sistema web de nutrición?

2. ¿Qué tipos de usuarios deberán tener acceso al sistema (administrador, médico, estudiante)?
3. ¿Qué información considera fundamental que registre un estudiante (peso, talla, historial médico, hábitos alimenticios)?
4. ¿Qué funcionalidades son indispensables para los médicos dentro de la plataforma?
5. ¿Qué tan importante considera la inclusión de un módulo de citas médicas en la solución?
6. ¿Le gustaría que el sistema contara con un foro motivacional donde los estudiantes puedan compartir experiencias y consejos?
7. ¿Qué tipo de reportes le gustaría recibir a nivel institucional (ejemplo: número de estudiantes atendidos, actividades más comunes, ¿tendencias de salud)?
8. ¿Considera necesario que el sistema esté disponible también en dispositivos móviles (celulares/tablets)?

- Metodología

Para el desarrollo del proyecto HealthyView: Sistema Web de Monitoreo Nutricional y Hábitos Saludables, se optó por utilizar la Metodología Crystal, específicamente la variante Crystal Clear, ya que se adapta de manera adecuada a equipos de trabajo pequeños y a proyectos de baja criticidad, como lo es el desarrollo de una plataforma académica de apoyo a estudiantes.

La metodología Crystal se caracteriza por su flexibilidad, su énfasis en la comunicación constante entre los integrantes del equipo y la importancia de entregar software funcional de manera continua.

- Especificación de requisitos

- Requisitos Funcionales Nominales

FN 1		Inicio de sesión
Descripción		El sistema contará con un apartado que permita un inicio de sesión personal, el cual el usuario tendrá acceso siempre y cuando se haya registrado
Importancia		Esencial
Validación	Medible	El inicio de sesión será por medio del ingreso de los datos usados en su registro, los cuales serán tanto el correo proporcionado así como su contraseña correspondiente. Cuando inicie sesión, solo mostrará lo que puede ver y hacer con respecto a su rol (ya sea paciente, médico o administrador). Dentro de este mismo apartado contará con una validación por contraseña, el cual impedirá el acceso al sistema si su contraseña no es válida y en caso de que sea válida, habrá un inicio de sesión exitoso.
	Alcanzable	Se realizará una validación que corrobore que los datos ingresados por el usuario sean los mismos que se encuentran registrados en la base de datos y así permitirá el acceso.
	Relevante	Al llevar a cabo este requisito en el sistema, se logrará un inicio de sesión exitosa que permita a cada usuario visualizar su información correspondiente al tipo de rol que desempeña (paciente, médico y administrador)

Tabla 2 RFN1 - Inicio de sesión

FN 2		Gestión de pacientes
Descripción		El sistema deberá registrar, editar, eliminar y consultar la información de pacientes, incluyendo datos personales, historial clínico y estado de salud. Los registros estarán disponibles para los médicos y administradores autorizados.
Importancia		Esencial
Validación	Medible	La validez se comprueba al realizar pruebas con credenciales correctas e incorrectas y verificar que únicamente los usuarios autorizados logren ingresar.
	Alcanzable	El requisito se podrá cumplir ya que el sistema contará con formularios y apartados específicos para capturar, actualizar y organizar los datos de los pacientes, de manera que su información se mantenga ordenada y disponible cuando se requiera.
	Relevante	La importancia radica en que protege la información sensible de los usuarios y evita accesos indebidos, asegurando que cada perfil utilice solo las funciones que le corresponden.

Tabla 3 RFN2 - Gestión de pacientes

FN 3		Gestión de médicos
Descripción		El sistema deberá permitir el registro y administración de perfiles de médicos, incluyendo datos profesionales, área de especialidad y disponibilidad para citas.
Importancia		Esencial
V a l i d e z	Medible	La validez se demuestra al registrar, modificar y eliminar perfiles médicos, verificando que la información se guarde en la base de datos y que esté correctamente asociada a las citas y recetas médicas.
	Alcanzable	Su implementación es posible mediante un módulo de administración que gestione la información del personal médico con formularios y tablas relacionadas.
	Relevante	Este requisito asegura la correcta asignación de citas y garantiza que los pacientes sean atendidos por médicos activos y registrados.

Tabla 4 RFN3 - Gestión de médicos

FN 4		Gestión de administradores
Descripción		El sistema deberá permitir la creación y administración de perfiles de administradores, quienes serán responsables de supervisar y controlar el correcto funcionamiento del sistema.
Importancia		Esencial
V a l i d e z	Medible	La comprobación se da cuando los administradores pueden ingresar y realizar acciones de control sin afectar la información de otros roles.
	Alcanzable	Se logra al establecer diferentes niveles de acceso dentro del sistema, asignando privilegios específicos a los administradores.
	Relevante	Este requisito garantiza que exista personal encargado del monitoreo y mantenimiento, promoviendo la estabilidad y el orden en la gestión del sistema.

Tabla 5 RFN4 - Gestión de administradores

FN 5		Gestión de perfiles de pacientes
Descripción		El sistema deberá permitir que los pacientes consulten y actualicen su perfil personal, incluyendo información básica y de contacto, que sea útil para su atención médica y el seguimiento de su bienestar.
Importancia		Esencial
V a l i d e z	Medible	Se valida al modificar los datos del perfil y comprobar que los cambios se reflejan en el sistema.
	Alcanzable	El cumplimiento se alcanza mediante una interfaz accesible que conecta al usuario con su perfil y permite editar información específica.
	Relevante	Garantiza que los datos de los pacientes estén actualizados y disponibles para los médicos.

Tabla 6 RFN5 - Gestión de perfiles

FN 6		Gestión de recetas médicas
Descripción		El sistema deberá permitir que los médicos registren, modifiquen y eliminen recetas médicas electrónicas, indicando los medicamentos, dosis y duración del tratamiento.
Importancia		Esencial
V a l i d e z	Medible	La verificación se da al registrar una receta y comprobar que quede vinculada al paciente correspondiente.
	Alcanzable	Se implementa mediante el diseño un formulario que guarde y organice las recetas de manera clara.
	Relevante	Este requisito facilita la prescripción médica y el seguimiento digital de los tratamientos de cada paciente.

Tabla 7 RFN6 - Gestión de recetas médicas

FN 7		Gestión de actividades saludables
Descripción		El sistema deberá permitir que los médicos recomienden actividades saludables (como ejercicio o cambios de alimentación) y que los pacientes puedan consultarlas y dar seguimiento a su cumplimiento.
Importancia		Esencial
V a l i d e z	Medible	Se valida cuando se asigna una actividad a un paciente y verifica que pueda visualizarla en su perfil.
	Alcanzable	El cumplimiento es posible al crear un módulo que vincule las recomendaciones del médico con las cuentas de los pacientes.
	Relevante	Fomenta la adopción de hábitos saludables y un acompañamiento activo por parte del profesional médico.

Tabla 8 RFN7 - Gestión de actividades saludables

FN 8		Gestión de citas médicas
Descripción		El sistema deberá permitir que los pacientes agenden, modifiquen y cancelen citas médicas, mostrando la disponibilidad de los médicos y evitando cruces de horarios.
Importancia		Esencial
V a l i d e z	Medible	La validez se observa cuando las citas se registran en el calendario del médico y del paciente sin generar cruces de horario.
	Alcanzable	Se puede implementar un control de horarios accesible desde la plataforma.
	Relevante	Optimiza la atención médica y evita problemas en la programación de citas.

Tabla 9 RFN8 - Gestión de citas médicas

FN 9		Asignación de recetas a pacientes
Descripción		El sistema deberá permitir que los médicos asignen recetas directamente a los pacientes, quedando almacenadas en su historial personal.
Importancia		Esencial
V a l i d e z	Medible	Se verifica cuando se registra una receta y que el paciente pueda consultarla en su expediente.
	Alcanzable	Es posible mediante la vinculación del módulo de recetas con los perfiles de pacientes.
	Relevante	Favorece el control de los tratamientos médicos y garantiza que el paciente reciba la prescripción correcta.

Tabla 10 RFN9 - Asignación de recetas a pacientes

FN 10		Registro de progreso de actividades
Descripción		El sistema deberá permitir que los pacientes registren sus avances en las actividades saludables recomendadas, de forma que los médicos puedan monitorear su cumplimiento.
Importancia		Esencial
V a l i d e z	Medible	Se confirma <u>al</u> ingresar registros de progreso y verificar que aparezcan en el historial del paciente.
	Alcanzable	Su implementación requiere de un apartado donde los pacientes registren sus avances.
	Relevante	Contribuye al seguimiento del esfuerzo y compromiso de los pacientes con su salud.

Tabla 11 RFN10 - Registro de progreso de actividades

FN 11		Registro de seguimientos clínicos
Descripción		El sistema deberá permitir a los médicos registrar el seguimiento clínico de cada paciente, guardando información sobre consultas previas, evolución de su estado y observaciones médicas.
Importancia		Esencial
V a l i d e z	Medible	Se comprueba al realizar varias consultas y visualizar que el historial muestra la evolución del paciente.
	Alcanzable	Su aplicación se logra mediante un apartado específico que almacene las consultas y observaciones médicas.
	Relevante	Es importante porque asegura la continuidad del tratamiento y la revisión del estado clínico del paciente.

Tabla 12 RFN11 - Registro de seguimientos clínicos

FN 12		Generación de reportes
Descripción		El sistema deberá generar reportes estadísticos sobre pacientes, médicos, citas y actividades, presentando información resumida que pueda ser consultada en pantalla o exportada.
Importancia		Esencial
V a l i d e z	Medible	Se constata al solicitar un reporte y verificar que muestre los datos solicitados.
	Alcanzable	El cumplimiento se logra al consultar y organizar la información que ya almacena y presentarla en diferentes formatos.
	Relevante	Proporciona información clave para la toma de decisiones y la evaluación del sistema.

Tabla 13 RFN12 - Generación de reportes

FN 13		Publicación de mensajes en foro
Descripción		El sistema deberá permitir que los pacientes publiquen mensajes en un foro motivacional, así como responder a publicaciones y dar retroalimentación a otros usuarios.
Importancia		Esencial
V a l i d e z	Medible	Se verifica al publicar mensajes y comprobar que otros usuarios puedan verlos y responder.
	Alcanzable	Su ejecución se logra al implementar un espacio de interacción dentro de la plataforma.
	Relevante	Fomenta la motivación y el apoyo entre los usuarios del sistema.

Tabla 14 RFN13 - Publicación de mensajes en foro

FN 14		Respaldo y restauración de la base de datos
Descripción		El sistema deberá permitir crear copias de seguridad automáticas y manuales de la información almacenada, así como restaurarlas en caso de pérdida o daño de datos. De esta manera se garantizará que la información pueda recuperarse y que el servicio continúe funcionando sin interrupciones graves.
Importancia		Esencial
V a l i d e z	Medible	La validez se confirma al generar un respaldo y posteriormente restaurarlo en un entorno de prueba, verificando que la información se conserve íntegra y accesible.
	Alcanzable	Se implementa con mecanismos sencillos que automaticen la creación de respaldos y que permitan recuperar la información cuando sea necesario.
	Relevante	Asegura la protección de los datos y la continuidad del sistema, evitando pérdidas de información crítica en situaciones de falla o error.

Tabla 15 RFN14 - Respaldo y restauración de la base de datos

- Requisitos Funcionales no Nominales

FF 1		Mensajes de error
Descripción		El sistema deberá mostrar mensajes claros cuando ocurra un error en la conexión o en el registro de información, indicando al usuario qué acción debe realizar.
Importancia		Esencial
Validación	Medible	Se <u>evalúa</u> al provocar errores de prueba y verificar que aparezca el mensaje adecuado.
	Alcanzable	Su implementación se logra con validaciones internas que detectan fallos y notifican al usuario.
	Relevante	Orienta al usuario y mejora la experiencia de uso.

Tabla 16 RFF1 - Mensaje de error

RFF 2		Errores de inicio de sesión
Descripción		El sistema deberá mostrar un mensaje de error cuando el usuario intente acceder con datos incorrectos, explicando que debe ingresar nuevamente sus credenciales.
Importancia		Esencial
Validación	Medible	Se verifica al introducir datos erróneos en el inicio de sesión y que el sistema muestre la alerta.
	Alcanzable	Es posible mediante validaciones que comparan las credenciales con los registros de la base de datos.
	Relevante	Previene confusiones y asegura la protección de la información.

Tabla 17 RFF2 - Error de inicio de sesión

RFF 3		Validación de campos
Descripción		El sistema deberá mostrar mensajes de validación cuando el usuario no complete información obligatoria o la ingrese en formato incorrecto.
Importancia		Esencial
Validación	Medible	Se confirma realizar pruebas como dejar campos vacíos en pruebas y verificar que el sistema los detecta.
	Alcanzable	Su cumplimiento se logra con reglas de validación integradas en los formularios.
	Relevante	Evita errores de registro y mantiene la calidad de la información.

Tabla 18 RFF3 - Validación de campos

- Requisitos de interfaz

IN 1		Usabilidad
Descripción		La interfaz del sistema deberá ser clara, sencilla e intuitiva, permitiendo que los usuarios comprendan las funciones sin necesidad de capacitación avanzada.
Importancia		Esencial
V a l i d e z	Medible	Se evalúa al realizar pruebas con usuarios y observar si logran navegar sin dificultades.
	Alcanzable	Es posible a través de un diseño de pantallas con botones, menús y mensajes claros.
	Relevante	Mejora la experiencia del usuario y promueve el uso continuo del sistema.

Tabla 19 RIN1 - Usabilidad

- Requisitos de calidad

CA 1		Diseño responsivo
Descripción		El sistema deberá adaptarse automáticamente a distintos dispositivos, como computadoras, tabletas y teléfonos móviles.
Importancia		Esencial
V a l i d e z	Medible	Se evalúa al acceder al sistema desde diferentes dispositivos y comprobar su correcto funcionamiento.
	Alcanzable	Es posible a través de un diseño que considere distintos tamaños de pantalla.
	Relevante	Permite que el sistema sea accesible desde cualquier entorno digital.

Tabla 20 RCA1 - Diseño responsivo

- Requisitos de evolución

EV 1		MVC (Modelo vista-controlador)
Descripción		El sistema deberá estar organizado en módulos independientes que faciliten su mantenimiento y futuras ampliaciones, siguiendo el enfoque de separación por capas.
Importancia		Esencial
V a l i d e z	Medible	Se demuestra al añadir un módulo nuevo sin afectar los ya existentes.
	Alcanzable	Al aplicar una planificación clara de las funciones del sistema.
	Relevante	Asegura la facilidad de actualización y escalabilidad.

Tabla 21 REV1 - MVC (Modelo vista-controlador)

- Requisitos de proyecto

PR 1		Tiempo de desarrollo
Descripción		El desarrollo del sistema se realizará en el periodo del 16 de septiembre al 15 de noviembre de 2025, dividido en fases de trabajo.
Importancia		Esencial
V a l i d e z	Medible	Se verifica al revisar el cronograma y los entregables de cada fase.
	Alcanzable	Es posible gracias a que el plan de actividades establece tiempos adecuados.
	Relevante	<u>Asegura</u> que el proyecto cumpla con los plazos académicos establecidos.

Tabla 22 RPR1 - Tiempo de desarrollo

- Requisitos de soporte

SO 1		Conexión a internet
Descripción		El sistema deberá contar con conexión estable a Internet para garantizar su uso correcto en todo momento.
Importancia		Esencial
V a l i d e z	Medible	Se evidencia al intentar usar el sistema sin conexión y comprobar que no funciona.
	Alcanzable	Se cumple considerando que el cliente cuenta con acceso a Internet.
	Relevante	Es indispensable al tratarse de un sistema web que opera en tiempo real.

Tabla 23 RSO1 - Conexión a internet

SO 2		Navegador compatibles
Descripción		El sistema deberá ser accesible desde navegadores actualizados como Google Chrome, Firefox y Microsoft Edge.
Importancia		Esencial
V a l i d e z	Medible	Se confirma al ejecutar el sistema en distintos navegadores y comprobar que mantiene sus funciones.
	Alcanzable	Se logra adaptando el código y las tecnologías utilizadas a los estándares de los navegadores actuales.
	Relevante	Garantiza el acceso para la mayoría de los usuarios sin restricciones técnicas.

Tabla 24 RSO2 – Navegador compatible

Capítulo III - Modelado del Sistema Administración / Investigación

En este capítulo se desarrolla el modelado del sistema HealthyView, presentando los diagramas y estructuras que describen el funcionamiento interno, la lógica de los módulos y la interacción entre sus componentes principales.

3.1 Diseño de requisitos funcionales.

A continuación, se presentan los diagramas de flujo correspondientes a cada uno de los Requisitos Funcionales Nominales (RFN) del sistema HealthyView. Estos diagramas ilustran de manera simple, clara y secuencial el funcionamiento interno de cada módulo, mostrando los procesos, decisiones y resultados esperados en la ejecución de cada función. Su propósito es facilitar la comprensión del comportamiento del sistema y servir como apoyo visual para el desarrollo y validación del proyecto.

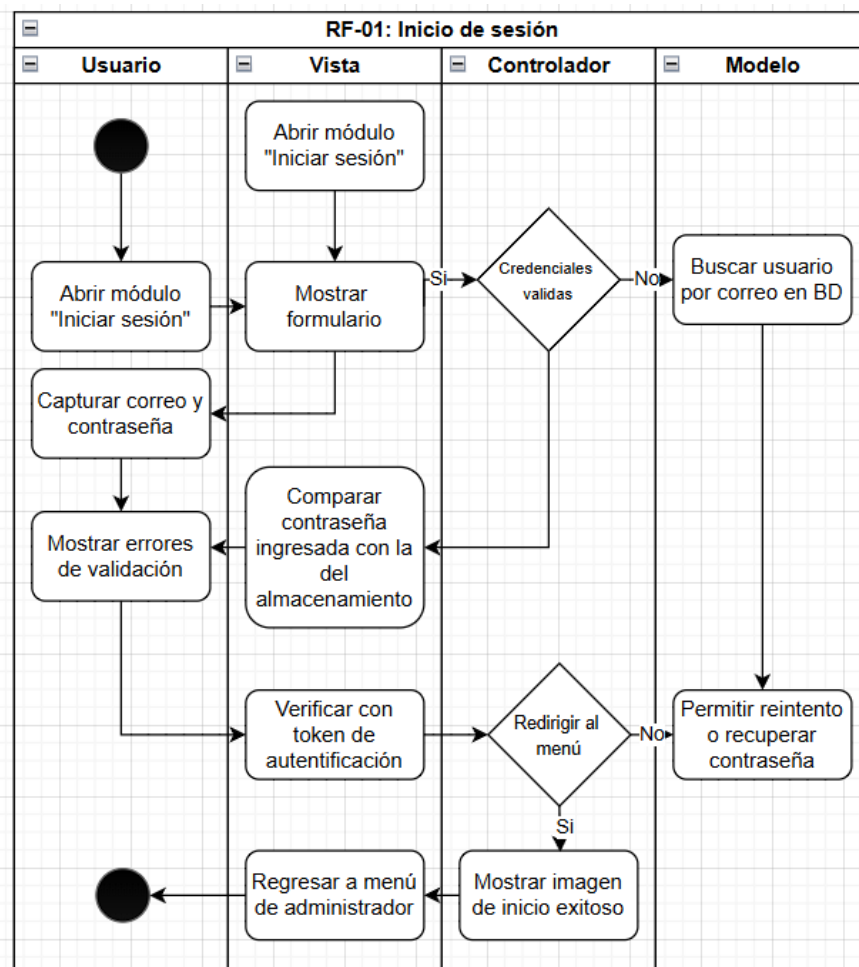


Ilustración 3 Diagrama de actividad RF-01: Inicio de sesión

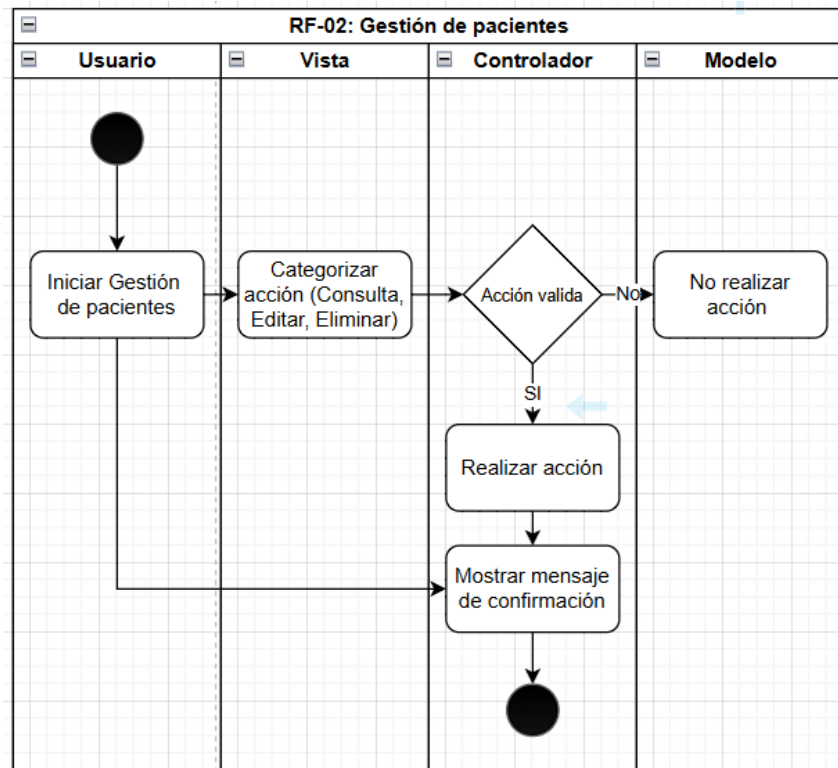


Ilustración 4 Diagrama de actividad RF-02: Gestión de pacientes

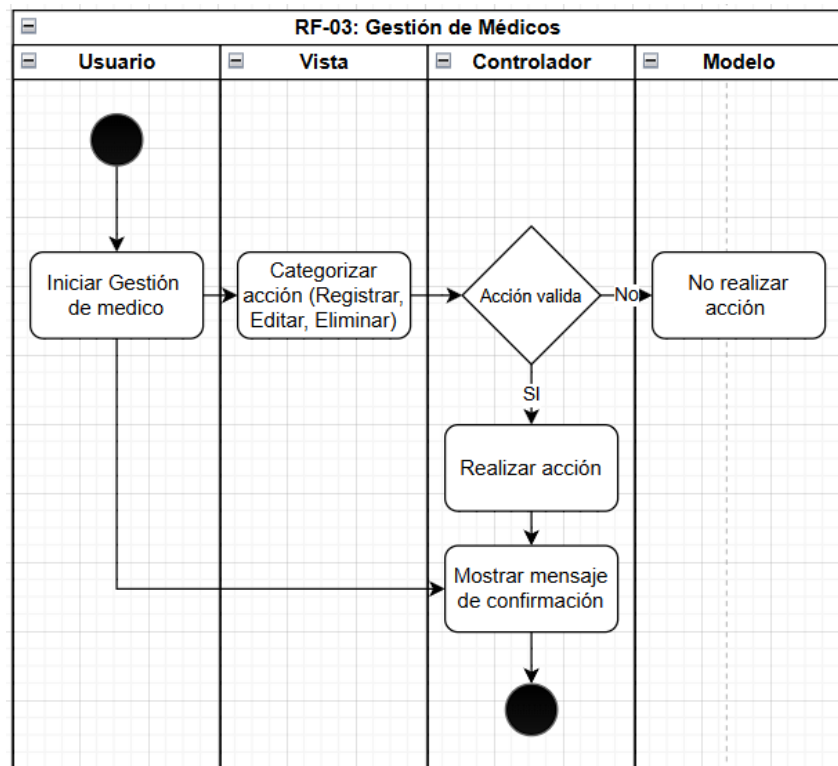


Ilustración 5 Diagrama de actividad RF-03: Gestión de Médicos

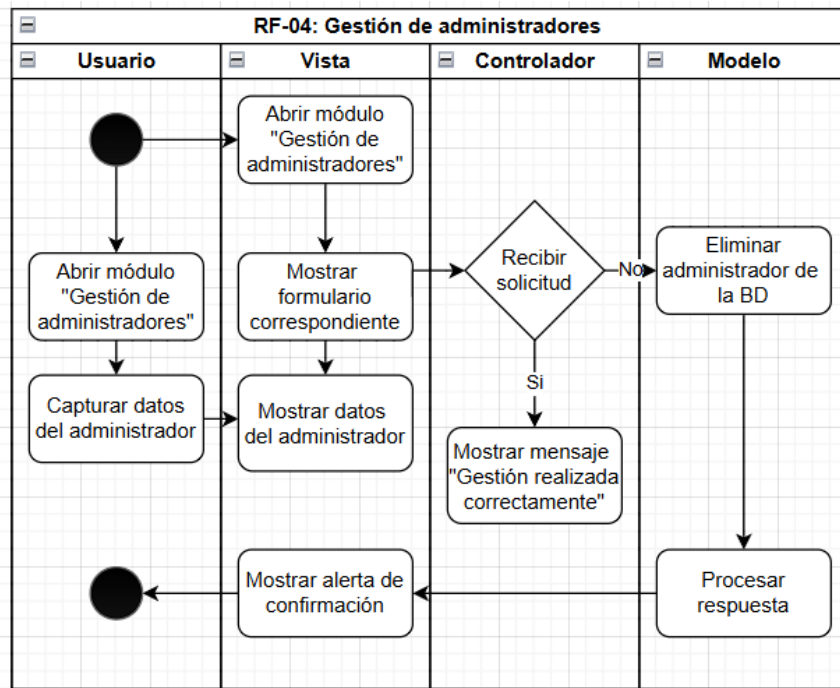


Ilustración 6 Diagrama de actividad RF-04: Gestión de administradores

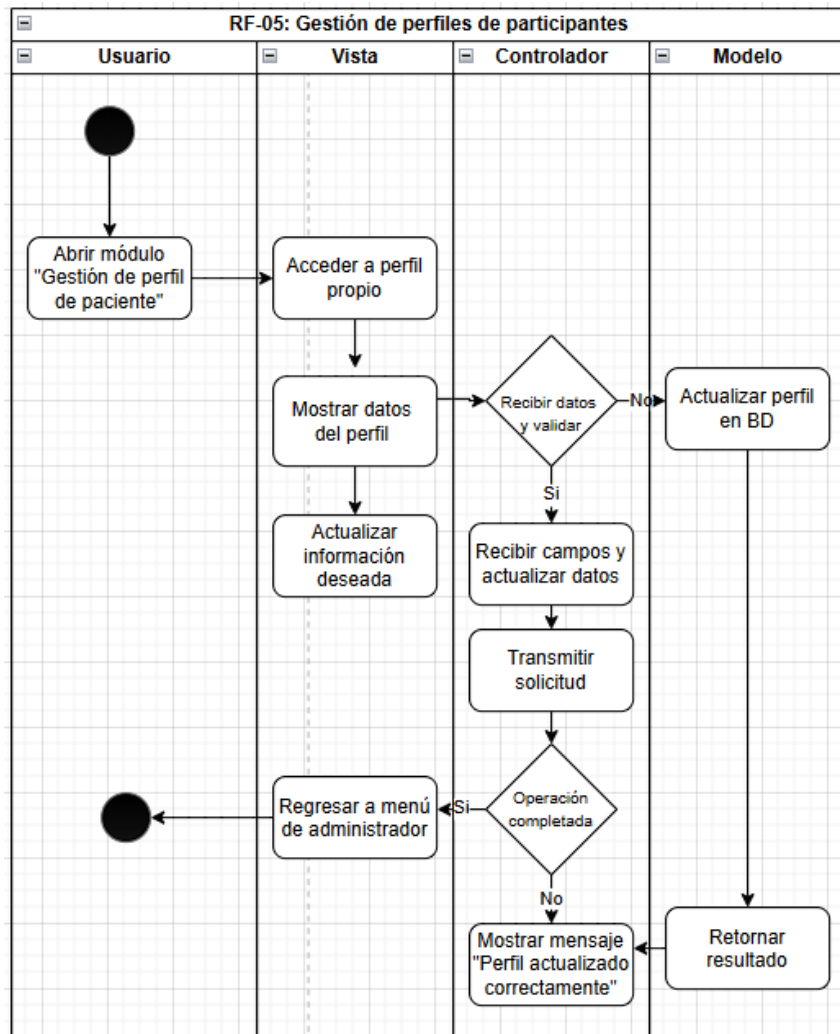


Ilustración 7 Diagrama de actividad RF-05: Gestión de perfiles de participantes

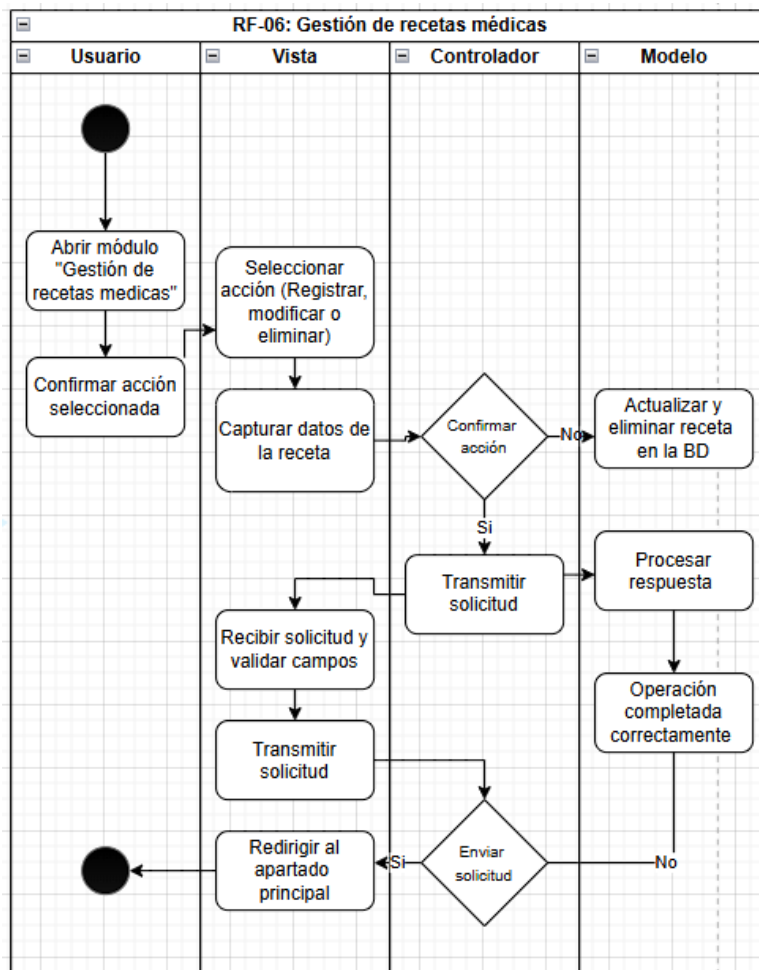


Ilustración 8 Diagrama de actividad RF-06: Gestión de recetas médicas

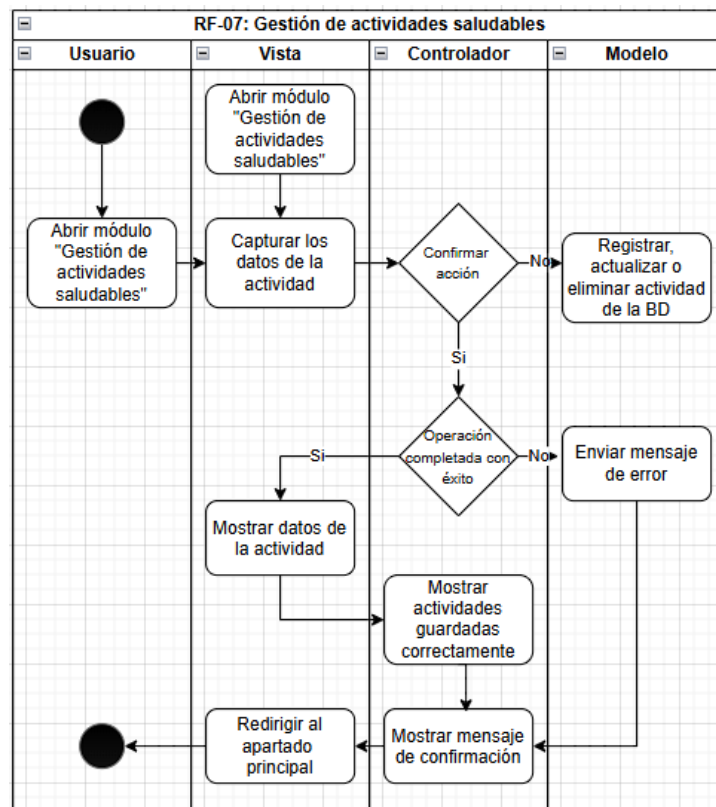


Ilustración 9 Diagrama de actividad RF-07: Gestión de actividades saludables

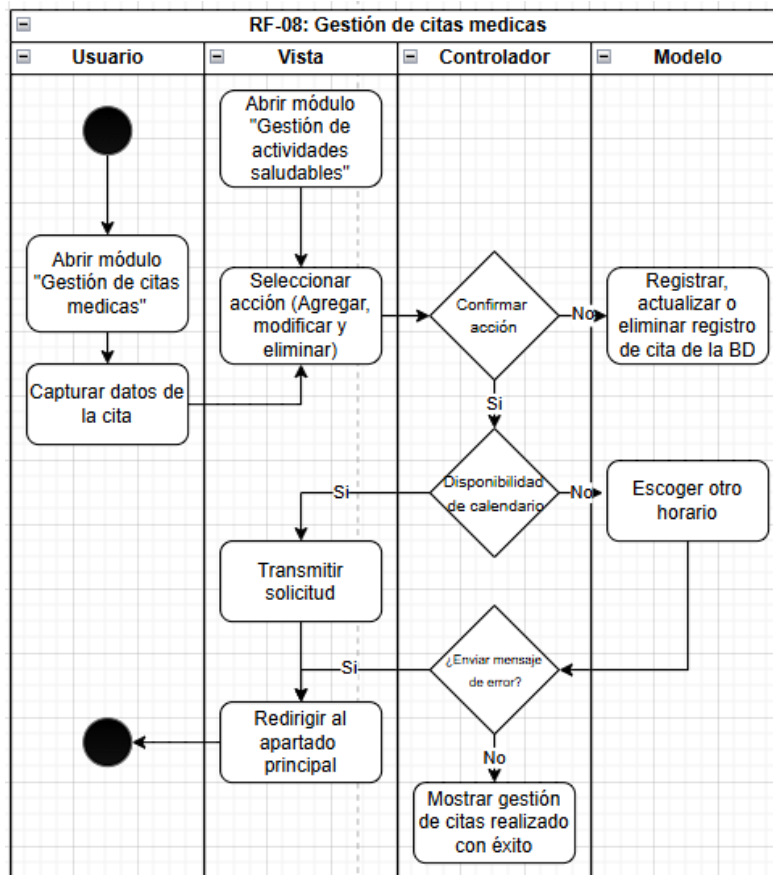


Ilustración 10 Diagrama de actividad RF-08: Gestión de citas medicas

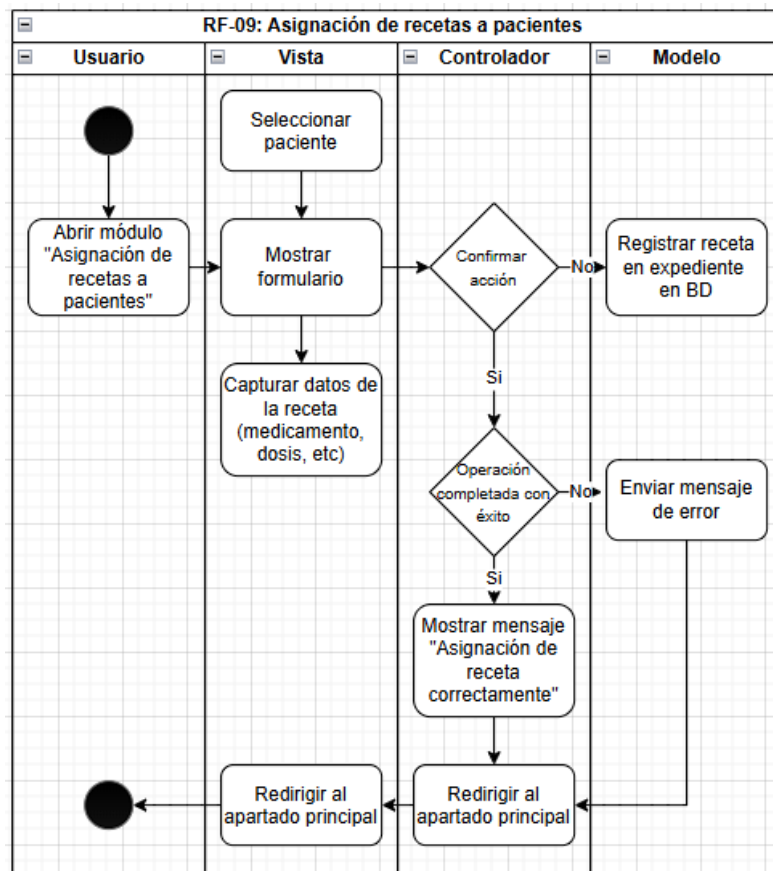


Ilustración 11 Diagrama de actividad RF-09: Asignación de recetas a pacientes

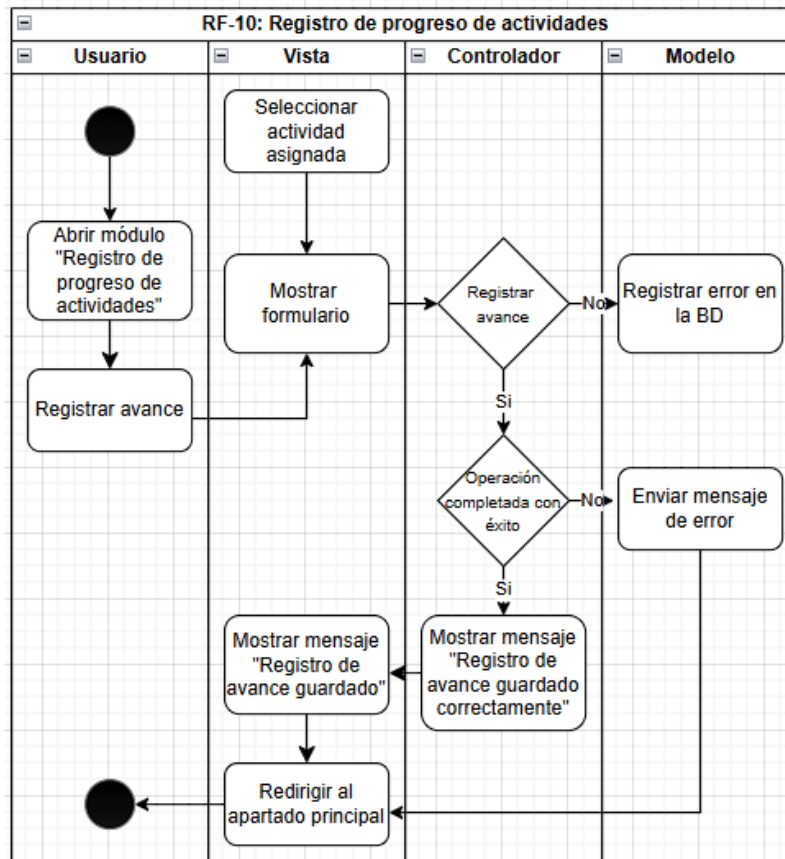


Ilustración 12 Diagrama de actividad RF-10: Registro de progreso de actividades

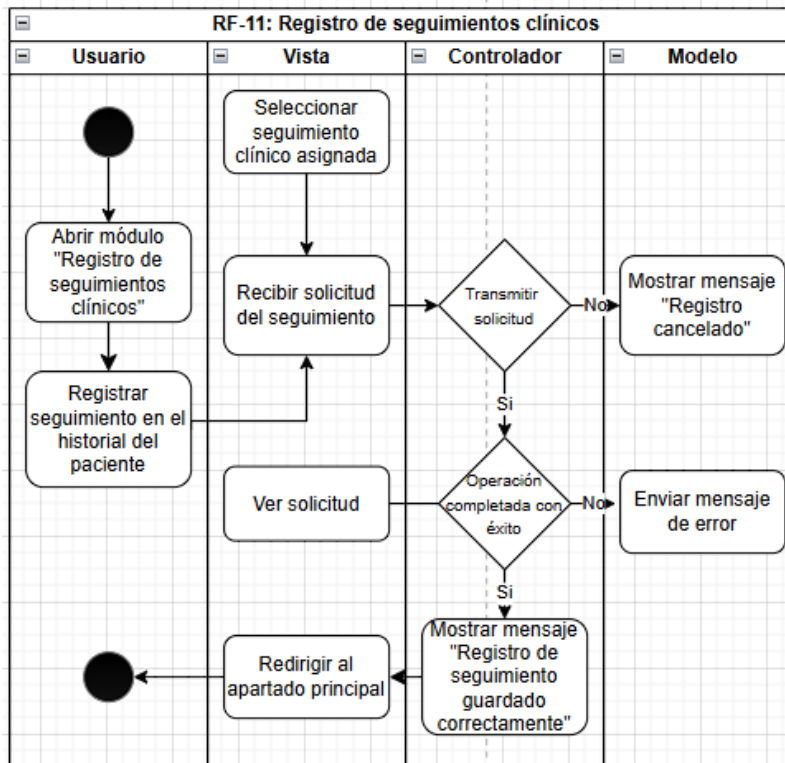


Ilustración 13 Diagrama de actividad RF-11: Registro de seguimientos clínicos

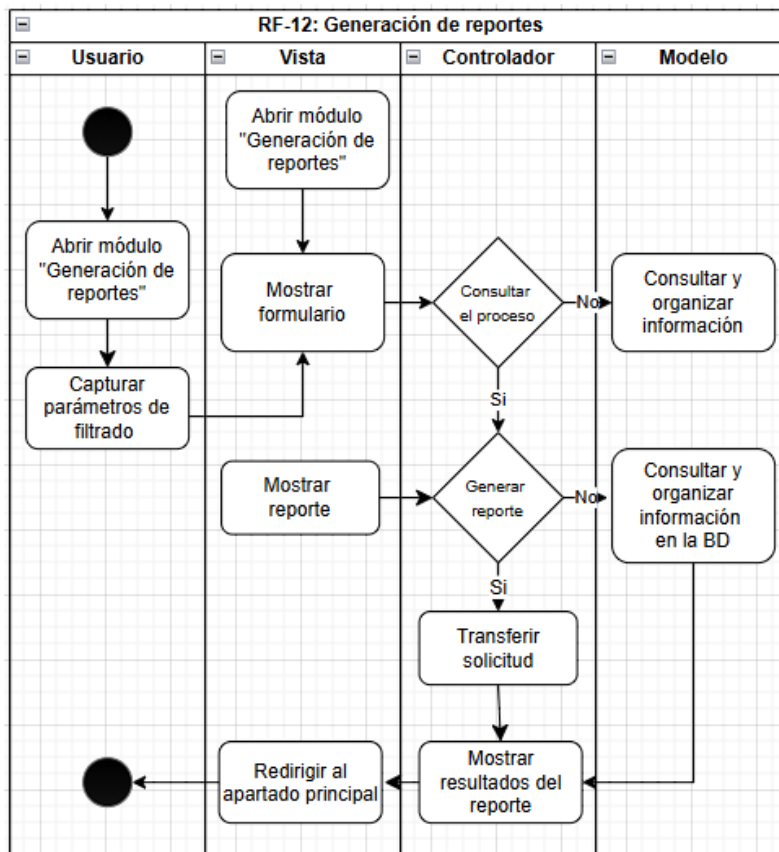


Ilustración 14 Diagrama de actividad RF-12: Generación de reportes

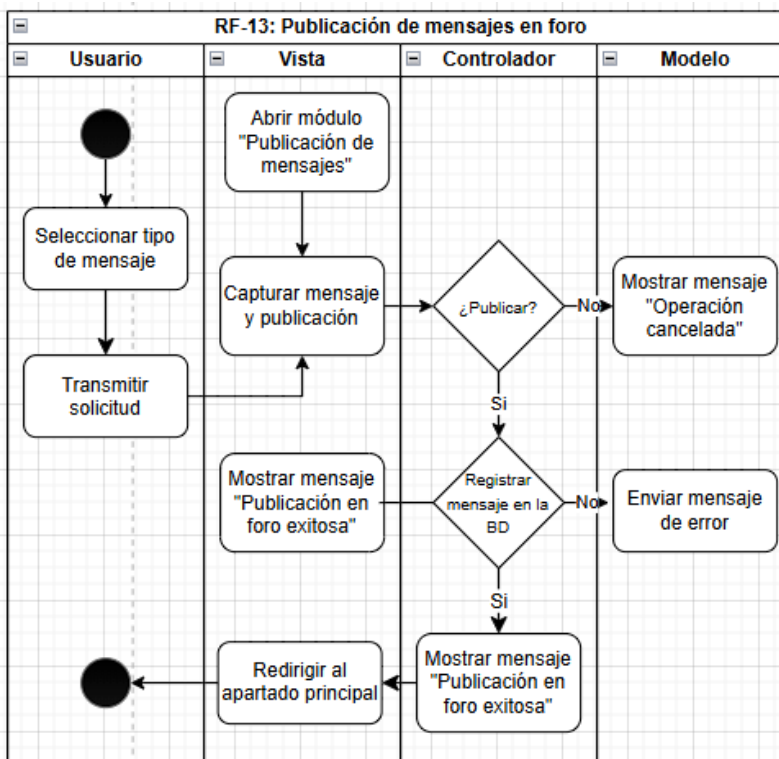


Ilustración 15 Diagrama de actividad RF-13: Publicación de mensajes en foro

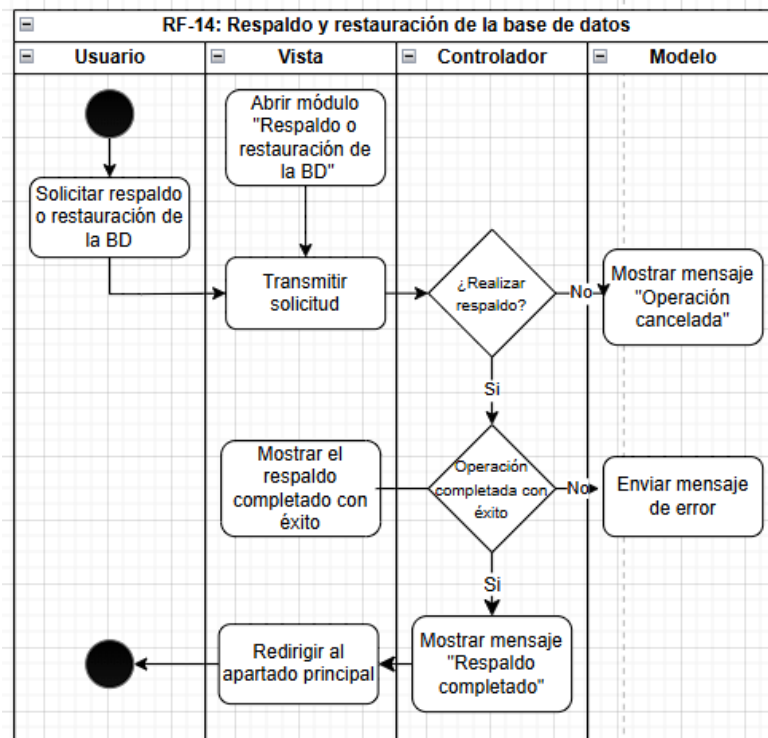


Ilustración 16 Diagrama de actividad RF-14: Respaldo y restauración de la base de datos

3.2 Arquitectura del sistema.

La arquitectura lógica del sistema HealthyView se basa en el patrón Modelo–Vista–Controlador (MVC), el cual permite una separación clara entre la lógica de negocio, la gestión de datos y la presentación visual. Este modelo garantiza una mejor organización del código, facilita el mantenimiento del sistema y permite realizar actualizaciones o ampliaciones sin afectar las demás capas.

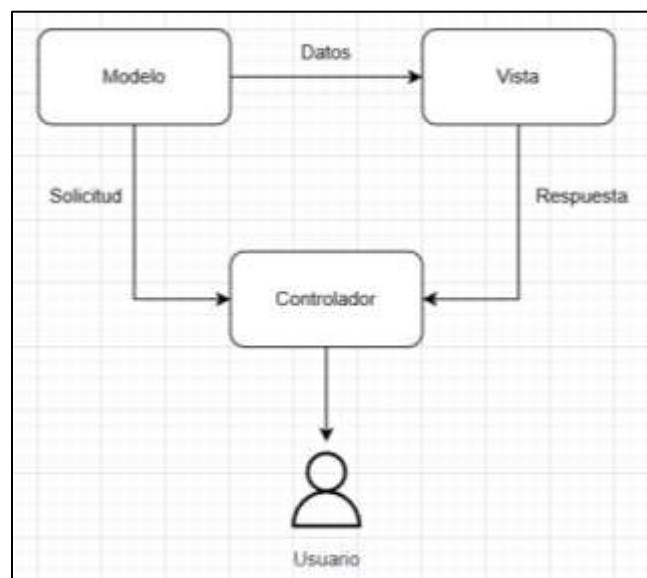


Ilustración 17 - Diagrama Modelo-Vista-Controlador (MVC)

En el diagrama se observa la relación entre los principales componentes del sistema:

- **Vista:** corresponde a la interfaz gráfica del usuario, representada por las pantallas, formularios y reportes desarrollados. Su función es mostrar la información proveniente del modelo y capturar las acciones del usuario (por ejemplo, iniciar sesión, registrar actividades, agendar citas o consultar reportes).
- **Controlador:** actúa como intermediario entre la vista y el modelo. Recibe las solicitudes del usuario desde la interfaz (Vista), las procesa y envía las instrucciones necesarias al modelo. Una vez obtenida la información, el controlador la devuelve a la vista en el formato adecuado para su presentación.
- **Modelo:** representa la capa lógica y de datos del sistema. Aquí se gestionan las operaciones sobre la base de datos —incluyendo pacientes, citas, seguimientos, actividades y recetas—, aplicando las reglas de negocio y garantizando la integridad de la información.

El flujo de comunicación se desarrolla de la siguiente manera: el usuario interactúa con la vista, la cual envía una solicitud al controlador. Este interpreta la acción y solicita los datos al modelo, que los procesa y devuelve una respuesta. Finalmente, el controlador actualiza la vista para mostrar los resultados al usuario.

Esta arquitectura permite que el sistema HealthyView mantenga una estructura modular, escalable y coherente, alineada con las buenas prácticas del desarrollo de software web.

3.3 Diseño de la interfaz de usuario.

En esta sección se presentan los diseños correspondientes a los principales componentes de la interfaz gráfica del sistema HealthyView, plataforma web

orientada al monitoreo nutricional y hábitos saludables para los usuarios. El propósito de esta fase es mostrar la estructura visual de las pantallas, ventanas y formularios que conforman la experiencia del usuario, priorizando criterios de usabilidad, claridad, accesibilidad y coherencia visual.

Cada boceto fue elaborado con base en los diagramas de actividades y la base de datos del proyecto, reflejando los flujos de interacción definidos en el análisis funcional. De este modo, el diseño de la interfaz no solo representa la parte estética del sistema, sino también la organización lógica de la información y las acciones que el usuario podrá realizar dentro de la plataforma.

1. Pantalla de inicio / Login

En esta interfaz se muestra el acceso principal al sistema HealthyView. Su diseño se centra en la simplicidad y la claridad visual, priorizando la usabilidad del usuario. El formulario de inicio de sesión contiene los campos esenciales: correo electrónico y contraseña, además de los enlaces para recuperar la contraseña o registrarse como nuevo usuario.

La disposición de los elementos facilita la interacción rápida y segura, mientras que la identidad visual colores suaves, tipografía legible y el logotipo institucional refuerza la coherencia estética del sistema. Esta pantalla constituye el punto de entrada al sistema, garantizando un acceso controlado a las funcionalidades internas.



2. Registro de nuevo paciente

Esta pantalla permite el registro de nuevos usuarios en el sistema HealthyView, brindando una interfaz clara y accesible para capturar la información personal y médica básica del paciente. El formulario incluye campos para nombre completo, correo electrónico, contraseña, género, peso, estatura y fecha de nacimiento, los cuales son necesarios para generar un perfil individualizado dentro de la base de datos del sistema.

El diseño mantiene una estructura ordenada y visualmente limpia, empleando colores suaves y tipografía legible para facilitar la lectura y la interacción. El botón principal “Crear cuenta” destaca por su contraste visual, guiando al usuario hacia la acción principal. Con esta interfaz se garantiza un proceso de registro eficiente, seguro y adaptado a los principios de usabilidad.



El boceto de la interfaz de registro de nuevo paciente en HealthyView presenta un diseño limpio y funcional. En la parte superior izquierda se encuentra el logo de HealthyView, que consiste en un corazón verde con una marca de verificación y el nombre de la aplicación. A la derecha, hay un enlace a 'Perfil' acompañado de un ícono de usuario. El título principal de la sección es 'Registro de nuevo paciente'. El formulario contiene los siguientes campos: un campo de texto para 'Nombre completo', un campo de texto para 'Correo electrónico', un campo de texto para 'Contraseña', un menú desplegable para 'Género' con la opción 'Seleccionar' visible, un campo de texto para 'Peso (kg)', un campo de texto para 'Estatura (cm)', y un botón azul destacado con el texto 'Crear cuenta'.

Ilustración 19 Boceto de interfaz del registro de nuevo paciente

3. Dashboard del paciente

Esta interfaz constituye la pantalla principal del usuario dentro del sistema HealthyView, donde se centraliza toda la información relevante sobre su progreso y estado de salud. El dashboard presenta un diseño estructurado con una barra lateral de navegación que permite acceder de forma rápida a los módulos de actividades de salud, seguimientos, recetas y citas programadas.

En el área principal se muestran indicadores clave como el peso actual, el nivel de progreso y la fecha de la próxima cita, además de un espacio dedicado a las metas personales del paciente, representadas mediante barras de avance. El diseño prioriza la claridad visual y la facilidad de acceso, permitiendo que el usuario interprete su estado y avance de manera intuitiva, alineándose con los principios de diseño centrado en el usuario y usabilidad.



Ilustración 20 Boceto de interfaz del dashboard del paciente

4. Gestión de actividades saludables

Esta interfaz permite al usuario administrar de forma sencilla sus actividades saludables dentro del sistema HealthyView, proporcionando un entorno organizado para el seguimiento de sus hábitos diarios. La pantalla presenta una estructura compuesta por un menú lateral de navegación que facilita el acceso a los demás módulos del sistema (seguimientos, recetas y citas), y un panel principal donde se muestra una tabla de actividades registradas.

Cada registro incluye el nombre de la actividad (por ejemplo, ejercicio, meditación, registro de comidas o sueño) y botones de acción que permiten editar o eliminar las entradas. Además, en la parte superior se dispone del botón “Agregar actividad”, que permite incorporar nuevas acciones de bienestar.

El diseño mantiene una estética limpia y coherente, priorizando la legibilidad y la usabilidad, de modo que el paciente pueda gestionar sus actividades de manera eficiente y sin distracciones visuales.



Ilustración 21 Boceto de interfaz de la gestión de actividades saludables

5. Seguimiento médico / monitoreo

Esta interfaz muestra el módulo de seguimiento médico del sistema HealthyView, donde se registran los avances del paciente con base en su peso, índice de masa corporal (IMC) y observaciones generales. La tabla central presenta los registros históricos de seguimiento, incluyendo la fecha, el peso, el IMC y notas relacionadas con la evolución del usuario.

En el costado derecho se incorpora un recuadro de comentarios del médico, espacio destinado para brindar recomendaciones o ajustes en el plan de salud. El diseño se caracteriza por su organización clara y su aspecto profesional, permitiendo una lectura rápida de los datos y favoreciendo la comunicación efectiva entre el paciente y el profesional de la salud.



Ilustración 22 Boceto de interfaz del monitoreo del médico

6. Gestión de citas

Esta interfaz muestra el módulo de gestión de citas médicas dentro del sistema HealthyView, donde el usuario puede visualizar, programar y administrar sus consultas. El diseño incluye una tabla central que presenta la información principal de cada cita: fecha, hora, motivo y médico asignado, junto con las opciones de editar o cancelar según sea necesario.

En la parte superior se incorpora el botón “Agendar nueva cita”, que facilita el registro de nuevas consultas médicas. El menú lateral mantiene la navegación coherente con el resto del sistema, permitiendo acceder fácilmente a los módulos de actividades, seguimientos y recetas. El enfoque de diseño se centra en la usabilidad y simplicidad, garantizando que el usuario gestione sus citas de manera eficiente y sin confusión visual.



Ilustración 23 Boceto de interfaz de la gestión de citas

7. Gestión de recetas

Esta interfaz corresponde al módulo de gestión de recetas del sistema HealthyView, el cual permite al paciente acceder a las recomendaciones alimenticias y planes nutricionales personalizados. La vista presenta una tabla organizada con los campos: nombre del platillo, ingredientes e instrucciones de preparación, acompañados por un botón de acción que permite descargar la receta en formato PDF para su consulta o impresión.

Además, se incluye un botón “Agregar receta”, destinado a que el médico o administrador pueda registrar nuevas indicaciones nutricionales dentro del sistema. El diseño mantiene la coherencia visual y la usabilidad, priorizando la legibilidad, la estructura jerárquica de la información y la facilidad de acceso para los distintos usuarios.

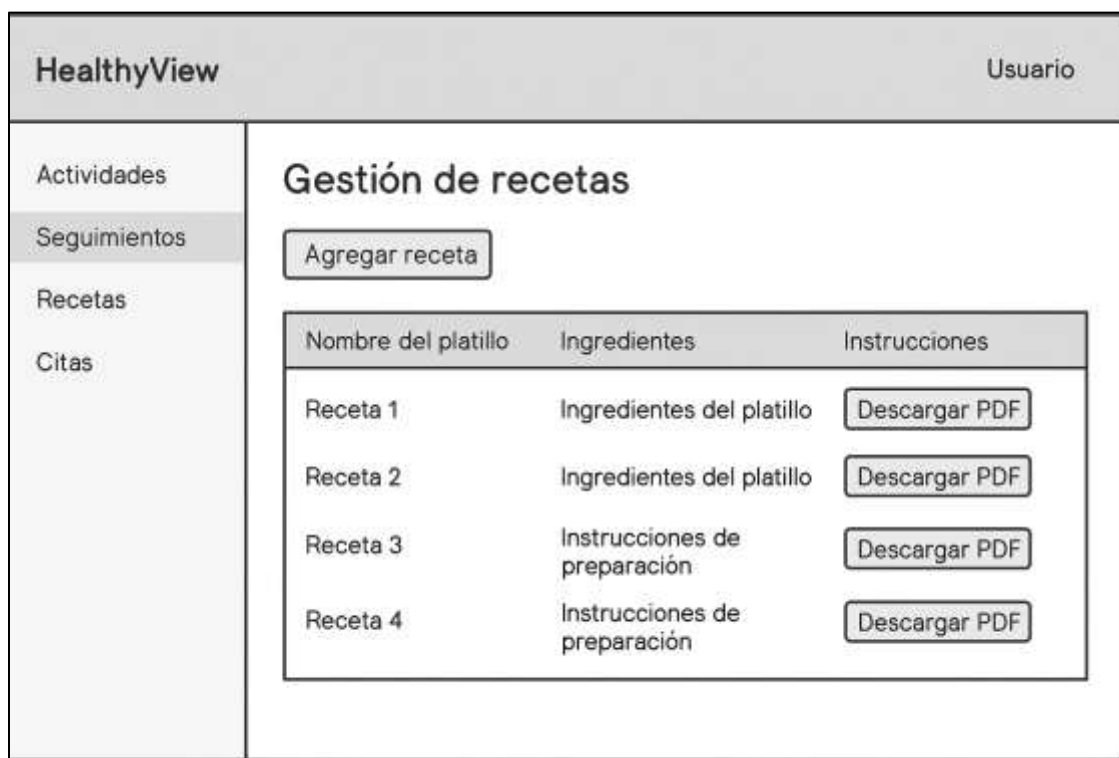


Ilustración 24 Boceto de interfaz de la gestión de recetas

8. Panel del administrador

Esta interfaz representa el panel principal del administrador dentro del sistema HealthyView, diseñado para la gestión general de usuarios y el control del funcionamiento del sistema. En la parte lateral se ubica un menú de navegación con acceso a las secciones de usuarios, reportes, actividades y base de datos, mientras que en el panel central se muestra una tabla con la información de los usuarios registrados, incluyendo su nombre, correo electrónico y rol dentro del sistema (paciente o doctor).

La interfaz incorpora un botón “Agregar usuario”, que permite registrar nuevos perfiles de manera rápida y eficiente. El diseño mantiene un estilo limpio y estructurado, orientado a la eficiencia administrativa y a la fácil localización de la información, garantizando la coherencia visual con el resto de las pantallas del sistema.

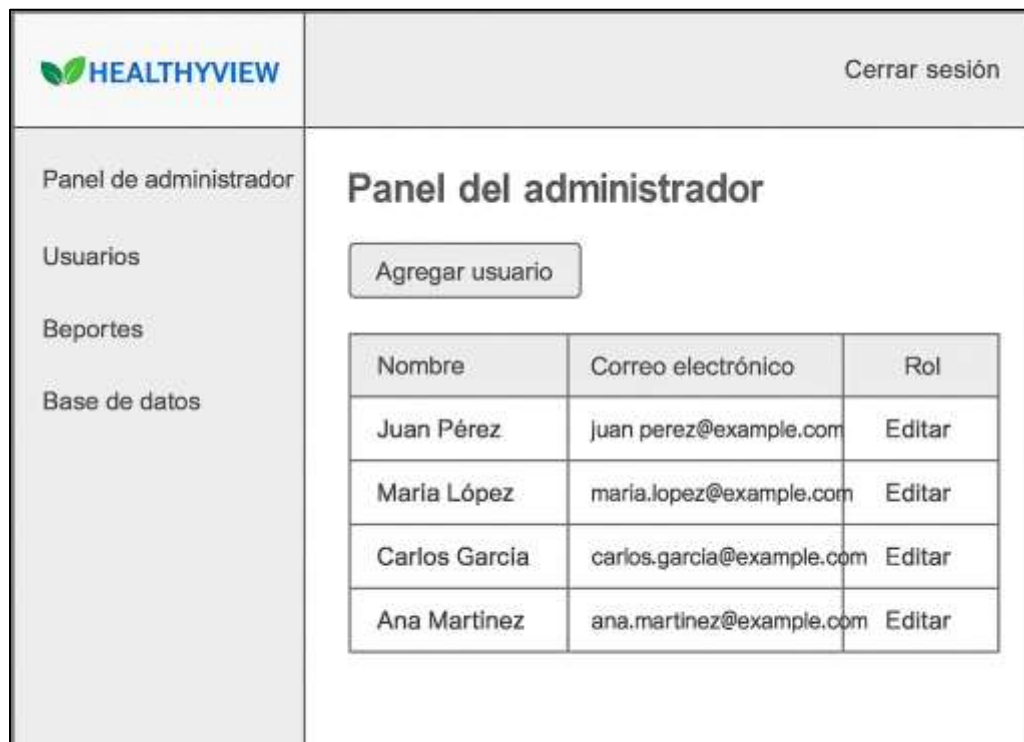


Ilustración 25 Boceto de interfaz del panel del administrador

9. Vista de impresión / reportes

Esta interfaz corresponde a la vista de impresión del sistema HealthyView, diseñada para generar reportes médicos y de progreso de los pacientes de forma clara y ordenada. El formato incluye los datos principales del usuario, como nombre, fecha, peso y nivel de progreso, además de un espacio destinado para las recomendaciones del médico y observaciones relevantes sobre el estado de salud.

Su diseño está optimizado para impresión o exportación en formato PDF, manteniendo una estructura limpia, legible y profesional. Esta vista permite documentar los avances del paciente y facilita la comunicación entre el profesional de salud y el usuario, alineándose con los estándares de presentación de información clínica y usabilidad del sistema.



HealthyView

Perfil

REPORTE

Nombre: _____

Fecha: _____

Peso: xx kg

Progreso: _____



RECOMENDACIONES

Ilustración 26 Boceto de interfaz de la vista de impresión de los reportes

10. Vista del doctor

Esta interfaz corresponde a la vista exclusiva del doctor dentro del sistema HealthyView, diseñada para facilitar el acceso a la información de los pacientes y optimizar el seguimiento médico. El panel principal presenta una tabla con el listado de pacientes registrados, mostrando su nombre, actividades, progreso y citas programadas, junto con botones que permiten visualizar los detalles individuales de cada uno.

El menú lateral ofrece un acceso rápido a las secciones de inicio, pacientes y perfil profesional, mientras que la parte superior incluye opciones para la navegación general y el cierre de sesión. El diseño está enfocado en la claridad, la organización y la eficiencia del trabajo médico, permitiendo que el profesional supervise múltiples casos de manera práctica, manteniendo una interfaz coherente con la identidad visual del sistema.



Ilustración 27 Boceto de interfaz de la vista desde el doctor

3.4 Diseño de base de datos.

- Modelo Relacional

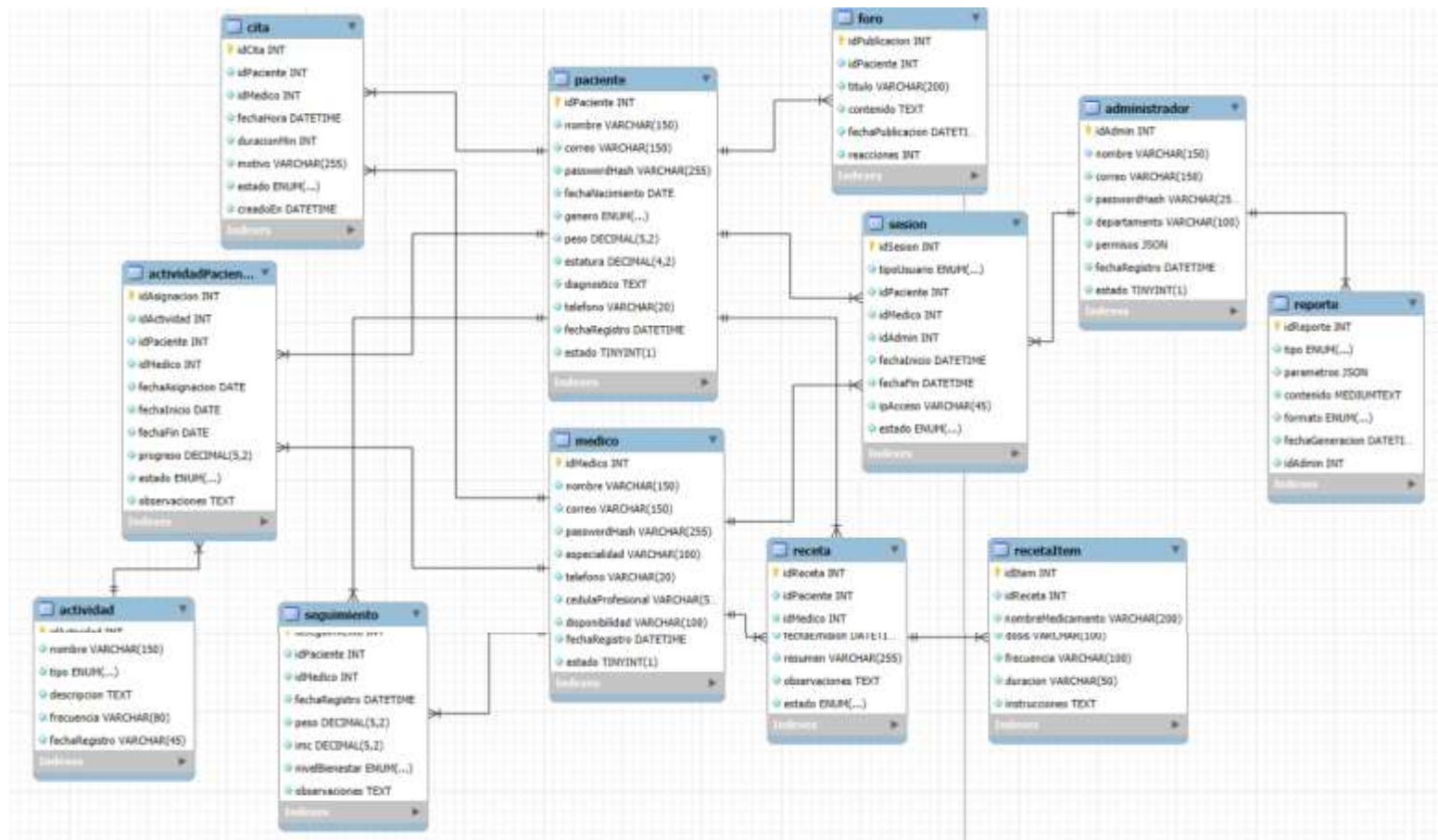


Ilustración 28 Modelo Relacional de nuestras tablas que conforman la base de datos (BD)

- Diccionario de datos

actividad				
Column name	DataType	PK	NN	Comment
idActividad	INT	✓	✓	Identificador de la actividad
nombre	VARCHAR(150)		✓	Nombre de la actividad
tipo	ENUM('Física', 'Alimentaria', 'Mental', 'Otro')		✓	Tipo de actividad que está realizando
descripcion	TEXT		✓	Descripción de la actividad
frecuencia	VARCHAR(80)		✓	Frecuencia con la que se hace la actividad
fechaRegistro	VARCHAR(45)		✓	Fecha en la que se registra

Tabla 25 Diccionario de datos de la tabla actividad

actividadPaciente				
Column name	DataType	PK	NN	Comment
idAsignacion	INT	✓	✓	Identificador de la actividad por paciente
idActividad	INT		✓	Identificador de la actividad
idPaciente	INT		✓	Identificador del paciente
idMedico	INT		✓	Identificador del medico
fechaAsignacion	DATE		✓	Fecha en la que se asigna la actividad
fechaInicio	DATE		✓	Fecha de inicio de las actividades
fechaFin	DATE		✓	Fecha en la que finalizan
progreso	DECIMAL(5,2)		✓	Progreso que se va observando
estado	ENUM('Activa', 'Completada', 'Cancelada')		✓	Se marca el estado en el que esta el paciente respecto a sus actividades
observaciones	TEXT		✓	Observaciones del paciente

Tabla 26 Diccionario de datos de la tabla actividadPaciente

administrador				
Column name	DataType	PK	NN	Comment
idAdmin	INT	✓	✓	Identificador del administrador
nombre	VARCHAR(150)		✓	Nombre del administrador
correo	VARCHAR(150)		✓	Correo del administrador
passwordHash	VARCHAR(255)		✓	Contraseña del administrador
departamento	VARCHAR(100)		✓	Departamento en el que está el administrador
permisos	JSON		✓	Permisos del administrador
fechaRegistro	DATETIME		✓	Fecha en el que se registró el administrador
estado	TINYINT(1)		✓	Estado en el que se encuentra el administrador

Tabla 27 Diccionario de datos de la tabla administrador

cita				
Column name	DataType	PK	NN	Comment
idCita	INT	✓	✓	Identificador de Citas
idPaciente	INT		✓	Identificador del paciente
idMedico	INT		✓	Identificador del medico
fechaHora	DATETIME		✓	Fecha y hora de la cita
duracionMin	INT		✓	Duración de la cita
motivo	VARCHAR(255)		✓	Motivo para realizar la cita
estado	ENUM('Programada', 'Completada', 'Cancelada', 'NoAsistida')		✓	Estado en el que se encuentra la cita
creadoEn	DATETIME		✓	Donde esta creada la cita

Tabla 28 Diccionario de datos de la tabla cita

foro				
Column name	DataType	PK	NN	Comment
idPublicacion	INT	✓	✓	Identificador de la publicación
idPaciente	INT		✓	Identificador del paciente
titulo	VARCHAR(200)		✓	Título de la publicación
contenido	TEXT		✓	Contenido de la publicación
fechaPublicacion	DATETIME		✓	Fecha en la que se hizo la publicación
reacciones	INT		✓	Reacciones que obtuvo la publicación

Tabla 29 Diccionario de datos de la tabla foro

medico				
Column name	DataType	PK	NN	Comment
idMedico	INT	✓	✓	Identificador del medico
nombre	VARCHAR(150)		✓	Nombre del medico
correo	VARCHAR(150)		✓	Correo del medico
passwordHash	VARCHAR(255)		✓	Contraseña del medico
especialidad	VARCHAR(100)		✓	Especialidad que tiene el medico
telefono	VARCHAR(20)		✓	Teléfono celular del medico
cedulaProfesional	VARCHAR(50)		✓	Cedula profesional del medico
disponibilidad	VARCHAR(100)		✓	Disponibilidad que tiene el medico
fechaRegistro	DATETIME		✓	Fecha en la que se registró el medico
estado	TINYINT(1)		✓	Estado en el que se encuentra el medico

Tabla 30 Diccionario de datos de la tabla medico

paciente				
Column name	DataType	PK	NN	Comment
idPaciente	INT	✓	✓	Identificador del paciente
nombre	VARCHAR(150)		✓	Nombre del paciente
correo	VARCHAR(150)		✓	Correo del paciente
passwordHash	VARCHAR(255)		✓	Contraseña del paciente
fechaNacimiento	DATE		✓	Fecha de nacimiento del paciente
genero	ENUM('Masculino', 'Femenino', 'Otro')		✓	Genero del paciente
peso	DECIMAL(5,2)		✓	Peso del paciente
estatura	DECIMAL(4,2)		✓	Estatura del paciente
diagnostico	TEXT		✓	Diagnóstico del paciente
telefono	VARCHAR(20)		✓	Teléfono celular del paciente
fechaRegistro	DATETIME		✓	Fecha de registro del paciente
estado	TINYINT(1)		✓	Estado actual del paciente

Tabla 31 Diccionario de datos de la tabla paciente

receta				
Column name	DataType	PK	NN	Comment
idReceta	INT	✓	✓	Identificador de la receta
idPaciente	INT		✓	Identificador del paciente
idMedico	INT		✓	Identificador del medico
fechaEmision	DATETIME		✓	Fecha en la que se emitió la receta
resumen	VARCHAR(255)		✓	Resumen de la receta
observaciones	TEXT		✓	Observaciones de la receta
estado	ENUM('Activa', 'Finalizada', 'Cancelada')		✓	Estado actual de la receta

Tabla 32 Diccionario de datos de la tabla receta

recetaltem				
Column name	DataType	PK	NN	Comment
idItem	INT	✓	✓	Identificador del contenido de la receta
idReceta	INT		✓	Identificador de la receta
nombreMedicamento	VARCHAR(200)		✓	Nombre del medicamento asignado
dosis	VARCHAR(100)		✓	Dosis del medicamento que fue asignado
frecuencia	VARCHAR(100)		✓	Frecuencia con la que se tiene que tomar
duracion	VARCHAR(50)		✓	Duración del medicamento
instrucciones	TEXT		✓	Instrucciones para tomar el medicamento

Tabla 33 Diccionario de datos de la tabla recetaltem

reporte				
Column name	DataType	PK	NN	Comment
idReporte	INT	✓	✓	Identificador del Reporte
tipo	ENUM('Pacientes', 'Citas', 'Actividades', 'General')		✓	Tipo de reporte que se va a emitir
parametros	JSON		✓	Parámetros para respetar del reporte
contenido	MEDIUMTEXT		✓	Contenido que habrá en el reporte
formato	ENUM('JSON', 'CSV', 'PDF')		✓	Formato asignado para el reporte
fechaGeneracion	DATETIME		✓	Fecha en la que se genero el reporte
idAdmin	INT		✓	Identificador del administrador

Tabla 34 Diccionario de datos de la tabla reporte

seguimiento				
Column name	DataType	PK	NN	Comment
idSeguimiento	INT	✓	✓	Identificador del seguimiento
idPaciente	INT		✓	Identificador del paciente
idMedico	INT		✓	Identificador del medico
fechaRegistro	DATETIME		✓	Fecha de registro de los seguimientos
peso	DECIMAL(5,2)		✓	Peso de paciente
imc	DECIMAL(5,2)		✓	Índice de masa corporal del paciente
nivelBienestar	ENUM('Excelente', 'Bueno', 'Regular', 'Malo')		✓	Nivel de bienestar en el que se considera que esta el paciente
observaciones	TEXT		✓	Observaciones del paciente

Tabla 35 Diccionario de datos de la tabla seguimiento

sesion				
Column name	DataType	PK	NN	Comment
idSesion	INT	✓	✓	Identificador de la sesión
tipoUsuario	ENUM('Paciente', 'Medico', 'Administrador')		✓	Tipo de usuario que está ingresando
idPaciente	INT		✓	Identificador del paciente
idMedico	INT		✓	Identificador del medico
idAdmin	INT		✓	Identificador del administrador
fechaInicio	DATETIME		✓	Fecha en la que inicio la sesión
fechaFin	DATETIME		✓	Fecha en la que finalizara
ipAcceso	VARCHAR(45)		✓	Confirmar el acceso
estado	ENUM('Activa', 'Cerrada', 'Fallida')		✓	Estado en el que se encuentra el usuario

Tabla 36 Diccionario de datos de la tabla sesion

- Prueba de la base de datos

Por el momento en cuanto a la base de datos únicamente tenemos la conexión y el llenado de tablas por lo que a continuación se muestra que efectivamente la base de datos ya tiene conexión y todas nuestras tablas con al menos 3 registros válidos.

#	Time	Action	Message
1	21:07:37	CREATE DATABASE IF NOT EXISTS healthyview	1 row(s) affected
2	21:07:37	USE healthyview	0 row(s) affected
3	21:07:37	CREATE TABLE paciente (idPaciente INT AUTO_INCREMENT PRIMARY KEY, -- identificador único del ...	0 row(s) affected
4	21:07:37	CREATE TABLE medico (idMedico INT AUTO_INCREMENT PRIMARY KEY, -- identificador único del ...	0 row(s) affected
5	21:07:37	CREATE TABLE administrador (idAdmin INT AUTO_INCREMENT PRIMARY KEY, -- identificador único...	0 row(s) affected
6	21:07:37	CREATE TABLE sesion (idSesion INT AUTO_INCREMENT PRIMARY KEY, -- identificador único de la ...	0 row(s) affected

Ilustración 29 Conexión a base BD

Tabla paciente:

```

32 * INSERT INTO paciente (nombre, correo, passwordHash, fechaNacimiento, genero, peso, estatura, diagnostico, telefono, estado)
33 VALUES
34 ('Ana García', 'ana.garcia@example.com', 'hash_ana_123', '1990-05-12', 'Femenino', 65.50, 1.65, 'Sin patologías importantes', '5512345678', 1),
35 ('Carlos López', 'carlos.lopez@example.com', 'hash_carlos_456', '1985-11-03', 'Masculino', 82.20, 1.78, 'Hipertensión leve', '5598765432', 1),
36 ('María Pérez', 'maria.perez@example.com', 'hash_maria_789', '1998-02-20', 'Femenino', 54.00, 1.60, 'Alergia a la penicilina', '5541122334', 1);
37
38 * SELECT * FROM paciente;

```

idPaciente	nombre	correo	passwordHash	fechaNacimiento	genero	peso	estatura	diagnostico	telefono	fechaRegistro	estado
1	Ana García	ana.garcia@example.com	hash_ana_123	1990-05-12	Femenino	65.50	1.65	Sin patologías importantes	5512345678	2025-10-11 12:34:06	1
2	Carlos López	carlos.lopez@example.com	hash_carlos_456	1985-11-03	Masculino	82.20	1.78	Hipertensión leve	5598765432	2025-10-11 12:34:06	1
3	María Pérez	maria.perez@example.com	hash_maria_789	1998-02-20	Femenino	54.00	1.60	Alergia a la penicilina	5541122334	2025-10-11 12:34:06	1

paciente 2 x

Output

Action Output

#	Time	Action	Message
12	21:07:37	CREATE TABLE seguimiento (idSeguimiento INT AUTO_INCREMENT PRIMARY KEY, --identificador	0 row(s) affected
13	21:07:37	CREATE TABLE foro (idPublicacion INT AUTO_INCREMENT PRIMARY KEY, --identificador de la pu...	0 row(s) affected
14	21:07:37	CREATE TABLE reporte (idReporte INT AUTO_INCREMENT PRIMARY KEY, --identificador único	0 row(s) affected
15	21:09:15	SELECT * FROM paciente LIMIT 0, 1000	0 row(s) returned
16	12:34:05	INSERT INTO paciente (nombre, correo, passwordHash, fechaNacimiento, genero, peso, estatura, diagn...	3 row(s) affected Records: 3 Duplicates: 0 Warnings: 0
17	12:34:14	SELECT * FROM paciente LIMIT 0, 1000	3 row(s) returned

Ilustración 30 Tabla paciente con registros exitosos

Tabla medico:

```

61 VALUES
62 ('Dr. Luis Sánchez', 'luis.sanchez@hospital.com', 'hash_luis_111', 'Nutriología', '5522233344', 'MED12345', 'Lun-Vie 9:00-15:00', 1),
63 ('Dra. Elena Ruiz', 'elena.ruiz@hospital.com', 'hash_elena_222', 'Medicina General', '5533344455', 'MED23456', 'Mar-Jue 10:00-18:00', 1),
64 ('Dr. Jorge Molina', 'jorge.molina@hospital.com', 'hash_jorge_333', 'Endocrinología', '5544455566', 'MED34567', 'Mié-Vie 8:00-14:00', 1);
65
66 * SELECT * FROM medico;
67
68 --

```

idMedico	nombre	correo	passwordHash	especialidad	telefono	cedulaProfesional	disponibilidad	fechaRegistro	estado
1	Dr. Luis Sánchez	luis.sanchez@hospital.com	hash_luis_111	Nutriología	5522233344	MED12345	Lun-Vie 9:00-15:00	2025-10-11 12:35:39	1
2	Dra. Elena Ruiz	elena.ruiz@hospital.com	hash_elena_222	Medicina General	5533344455	MED23456	Mar-Jue 10:00-18:00	2025-10-11 12:35:39	1
3	Dr. Jorge Molina	jorge.molina@hospital.com	hash_jorge_333	Endocrinología	5544455566	MED34567	Mié-Vie 8:00-14:00	2025-10-11 12:35:39	1

medico 3 x

Output

Action Output

#	Time	Action	Message
1	12:35:39	INSERT INTO medico (nombre, correo, passwordHash, especialidad, telefono, cedulaProfesional, disponibilidad...	3 row(s) affected Records: 3 Duplicates: 0 Warnings: 0
2	12:35:47	SELECT * FROM medico LIMIT 0, 1000	3 row(s) returned

Ilustración 31 Tabla medico con registros exitosos

Tabla administrador:

```

106 * INSERT INTO administrador (nombre, correo, passwordHash, departamento, permisos, estado)
107 VALUES
108 ('Admin Central', 'admin.central@example.com', 'hash_admin_1', 'Soporte', JSON_ARRAY('usuarios', 'reportes'), 1),
109 ('Admin RRHH', 'admin.rrhh@example.com', 'hash_admin_2', 'Recursos Humanos', JSON_ARRAY('empleados'), 1),
110 ('Admin IT', 'admin.it@example.com', 'hash_admin_3', 'TI', JSON_ARRAY('sistemas', 'backups'), 1);
111
112 * SELECT * FROM administrador;

```

idAdmin	nombre	correo	passwordHash	departamento	permisos	fechaRegistro	estado
1	Admin Central	admin.central@example.com	hash_admin_1	Soporte	['usuarios', 'reportes']	2025-10-11 12:41:40	1
2	Admin RRHH	admin.rrhh@example.com	hash_admin_2	Recursos Humanos	['empleados']	2025-10-11 12:41:40	1
3	Admin IT	admin.it@example.com	hash_admin_3	TI	['sistemas', 'backups']	2025-10-11 12:41:40	1

Administrador4 x

Output

Action Output

#	Time	Action	Message
1	12:41:40	INSERT INTO administrador (nombre, correo, passwordHash, departamento, permisos, estado) VALUES (Admin...	3 row(s) affected Records: 3 Duplicates: 0 Warnings: 0
2	12:41:43	SELECT * FROM administrador LIMIT 0, 1000	3 row(s) returned

Ilustración 32 Tabla administrador con registros exitosos

Tabla sesion:

```

116 * INSERT INTO sesion (tipoUsuario, idPaciente, idMedico, idAdmin, fechaInicio, fechaFin, ipAcceso, estado)
117 VALUES
118 ('Paciente', 1, NULL, NULL, NOW(), NULL, '192.168.1.10', 'Activa'),
119 ('Medico', NULL, 1, NULL, NOW(), NOW(), '192.168.1.11', 'Cerrada'),
120 ('Administrador', NULL, NULL, 1, NOW(), NULL, '192.168.1.12', 'Activa');
121
122 * SELECT * FROM sesion;

```

idSesion	tipoUsuario	idPaciente	idMedico	idAdmin	fechaInicio	fechaFin	ipAcceso	estado
1	Paciente	1			2025-10-11 12:42:49		192.168.1.10	Activa
2	Medico		1		2025-10-11 12:42:49	2025-10-11 12:42:49	192.168.1.11	Cerrada
3	Administrador			1	2025-10-11 12:42:49		192.168.1.12	Activa

sesion5 x

Output

Action Output

#	Time	Action	Message
1	12:42:49	INSERT INTO sesion (tipoUsuario, idPaciente, idMedico, idAdmin, fechaInicio, fechaFin, ipAcceso, estado) VAL...	3 row(s) affected Records: 3 Duplicates: 0 Warnings: 0
2	12:42:52	SELECT * FROM sesion LIMIT 0, 1000	3 row(s) returned

Ilustración 33 Tabla sesion con registros exitosos

Tabla receta:

```

143 * INSERT INTO receta (idPaciente, idMedico, fechaEmision, resumen, observaciones, estado)
144 VALUES
145 (1, 1, NOW(), 'Control de peso - inicio', 'Recomendaciones nutricionales básicas', 'Activa'),
146 (2, 2, NOW(), 'Chequeo general', 'Solicitar análisis sanguíneo', 'Activa'),
147 (3, 3, NOW(), 'Control metabólico', 'Ajustar dosis de suplementos', 'Activa');
148
149 * SELECT * FROM receta;

```

idReceta	idPaciente	idMedico	fechaEmision	resumen	observaciones	estado
1	1	1	2025-10-11 12:44:08	Control de peso - inicio	Recomendaciones nutricionales básicas	Activa
2	2	2	2025-10-11 12:44:08	Chequeo general	Solicitar análisis sanguíneo	Activa
3	3	3	2025-10-11 12:44:08	Control metabólico	Ajustar dosis de suplementos	Activa

receta E x

Output

Action Output

#	Time	Action	Message
1	12:44:08	INSERT INTO receta (idPaciente, idMedico, fechaEmision, resumen, observaciones, estado) VALUES (1, 1, NOW(), 'Control de peso - inicio', 'Recomendaciones nutricionales básicas', 'Activa');	3 row(s) affected Records: 3 Duplicates: 0 Warnings: 0
2	12:44:12	SELECT * FROM receta LIMIT 0, 1000	3 row(s) returned

Ilustración 34 Tabla receta con registros exitosos

Tabla recetItem:

```

169 * INSERT INTO recetItem (idReceta, nombreMedicamento, dosis, frecuencia, duracion, instrucciones)
170 VALUES
171 (1, 'Multivitamínico', '1 cápsula', '1 vez al día', '30 días', 'Tomar por la mañana con alimento'),
172 (2, 'Paracetamol', '500 mg', 'Cada 8 horas', '5 días', 'Si hay dolor o fiebre'),
173 (3, 'Omega-3', '1000 mg', '1 vez al día', '60 días', 'Tomar con comida');
174
175 * SELECT * FROM recetItem;

```

idItem	idReceta	nombreMedicamento	dosis	frecuencia	duracion	instrucciones
1	1	Multivitamínico	1 cápsula	1 vez al día	30 días	Tomar por la mañana con alimento
2	2	Paracetamol	500 mg	Cada 8 horas	5 días	Si hay dolor o fiebre
3	3	Omega-3	1000 mg	1 vez al día	60 días	Tomar con comida

recetItem 7. x:

Output

Action Output

#	Time	Action	Message
1	12:44:59	INSERT INTO recetItem (idReceta, nombreMedicamento, dosis, frecuencia, duracion, instrucciones) VALUES (1, 'Multivitamínico', '1 cápsula', '1 vez al día', '30 días', 'Tomar por la mañana con alimento');	3 row(s) affected Records: 3 Duplicates: 0 Warnings: 0
2	12:45:02	SELECT * FROM recetItem LIMIT 0, 1000	3 row(s) returned

Ilustración 35 Tabla recetItem con registros exitosos

Tabla actividad:

```

193 * INSERT INTO actividad (nombre, tipo, descripcion, frecuencia)
194 VALUES
195 ('Caminata diaria', 'Física', 'Caminata de 30 minutos a paso moderado', 'Diaria'),
196 ('Plan de comidas saludable', 'Alimentaria', 'Plan semanal con calorías controladas', 'Semanal'),
197 ('Meditación guiada', 'Mental', 'Sesiones de 15 minutos para control de estrés', 'Diaria');
198
199 * SELECT * FROM actividad;

```

idActividad	nombre	tipo	descripcion	frecuencia	fechaRegistro
1	Caminata diaria	Física	Caminata de 30 minutos a paso moderado	Diaria	2025-10-11 12:46:00
2	Plan de comidas saludable	Alimentaria	Plan semanal con calorías controladas	Semanal	2025-10-11 12:46:00
3	Meditación guiada	Mental	Sesiones de 15 minutos para control de estrés	Diaria	2025-10-11 12:46:00

Output

Action Output

#	Time	Action	Message
1	12:46:00	INSERT INTO actividad (nombre, tipo, descripcion, frecuencia) VALUES ('Caminata diaria', 'Física', 'Caminata d...	3 row(s) affected Records: 3 Duplicates: 0 Warnings: 0
2	12:46:03	SELECT * FROM actividad LIMIT 0, 1000	3 row(s) returned

Ilustración 36 Tabla actividad con registros exitosos

Tabla actividadPaciente:

```

224 * INSERT INTO actividadPaciente (idActividad, idPaciente, idMedico, fechaAsignacion, fechaInicio, fechaFin, progreso, estado, observaciones)
225 VALUES
226 (1, 1, 1, '2025-10-01', '2025-10-02', NULL, 10.00, 'Activa', 'Empezó bien'),
227 (2, 2, 2, '2025-10-03', NULL, NULL, 0.00, 'Activa', 'Programada para la próxima semana'),
228 (3, 3, NULL, '2025-10-05', NULL, NULL, 0.00, 'Activa', 'Paciente la hará en casa');
229
230 * SELECT * FROM actividadPaciente;

```

idAsignacion	idActividad	idPaciente	idMedico	fechaAsignacion	fechaInicio	fechaFin	progreso	estado	observaciones
1	1	1	1	2025-10-01	2025-10-02		10.00	Activa	Empezó bien
2	2	2	2	2025-10-03			0.00	Activa	Programada para la próxima semana
3	3	3		2025-10-05			0.00	Activa	Paciente la hará en casa

Output

Action Output

#	Time	Action	Message
1	12:46:40	INSERT INTO actividadPaciente (idActividad, idPaciente, idMedico, fechaAsignacion, fechaInicio, fechaFin, pr...	3 row(s) affected Records: 3 Duplicates: 0 Warnings: 0
2	12:46:43	SELECT * FROM actividadPaciente LIMIT 0, 1000	3 row(s) returned

Ilustración 37 Tabla actividadPaciente con registros exitosos

Tabla cita:

```

252 * INSERT INTO cita (idPaciente, idMedico, fechaHora, duracionMin, motivo, estado)
253 VALUES
254 (1, 1, '2025-10-10 09:30:00', 30, 'Consulta inicial nutrición', 'Programada'),
255 (2, 2, '2025-10-11 11:00:00', 40, 'Revisión general', 'Programada'),
256 (3, 3, '2025-10-12 08:30:00', 30, 'Control endocrinológico', 'Programada');
257
258 * SELECT * FROM cita;

```

	idCita	idPaciente	idMedico	fechaHora	duracionMin	motivo	estado	creadoEn
1	1	1	1	2025-10-10 09:30:00	30	Consulta inicial nutrición	Programada	2025-10-11 12:47:25
2	2	2	2	2025-10-11 11:00:00	40	Revisión general	Programada	2025-10-11 12:47:25
3	3	3	3	2025-10-12 08:30:00	30	Control endocrinológico	Programada	2025-10-11 12:47:25

Output

#	Time	Action	Message
1	12:47:25	INSERT INTO cita (idPaciente, idMedico, fechaHora, duracionMin, motivo, estado) VALUES (1, 1, '2025-10-10 09:30:00', 30, 'Consulta inicial nutrición', 'Programada')	3 row(s) affected Records: 3 Duplicates: 0 Warnings: 0
2	12:47:26	SELECT * FROM cita LIMIT 0, 1000	3 row(s) returned

Ilustración 38 Tabla cita con registros exitosos

Tabla seguimiento:

```

280 * INSERT INTO seguimiento (idPaciente, idMedico, fechaRegistro, peso, imc, nivelBienestar, observaciones)
281 VALUES
282 (1, 1, NOW(), 65.50, 24.08, 'Bueno', 'Paciente en seguimiento mensual'),
283 (2, 2, NOW(), 82.20, 25.98, 'Regular', 'Recomendar actividad física'),
284 (3, 3, NOW(), 54.00, 21.09, 'Excelente', 'Paciente reporta mejora');
285
286 * SELECT * FROM seguimiento;

```

	idSeguimiento	idPaciente	idMedico	fechaRegistro	peso	imc	nivelBienestar	observaciones
1	1	1	1	2025-10-11 12:48:10	65.50	24.08	Bueno	Paciente en seguimiento mensual
2	2	2	2	2025-10-11 12:48:10	82.20	25.98	Regular	Recomendar actividad física
3	3	3	3	2025-10-11 12:48:10	54.00	21.09	Excelente	Paciente reporta mejora

Output

#	Time	Action	Message
1	12:48:10	INSERT INTO seguimiento (idPaciente, idMedico, fechaRegistro, peso, imc, nivelBienestar, observaciones) VALUES (1, 1, NOW(), 65.50, 24.08, 'Bueno', 'Paciente en seguimiento mensual')	3 row(s) affected Records: 3 Duplicates: 0 Warnings: 0
2	12:48:13	SELECT * FROM seguimiento LIMIT 0, 1000	3 row(s) returned

Ilustración 39 Tabla seguimiento con registros exitosos

Tabla foro:

```

305 * INSERT INTO foro (idPaciente, titulo, contenido, reacciones)
306 VALUES
307 (1, 'Comencé a caminar', 'Hoy empecé la caminata diaria y me siento con más energía', 5),
308 (2, 'Duda sobre dieta', '¿Puedo comer frutas antes de dormir?', 2),
309 (3, NULL, 'Compartiendo mi progreso semanal: bajé 1.5 kg', 8);
310
311 * SELECT * FROM foro;
312

```

idPublicacion	idPaciente	titulo	contenido	fechaPublicacion	reacciones
1	1	Comencé a caminar	Hoy empecé la caminata diaria y me siento con ...	2025-10-11 12:49:20	5
2	2	Duda sobre dieta	¿Puedo comer frutas antes de dormir?	2025-10-11 12:49:20	2
3	3	Compartiendo mi progreso semanal: bajé 1.5 kg		2025-10-11 12:49:20	8

Output

Action Output

#	Time	Action	Message
1	12:49:20	INSERT INTO foro (idPaciente, titulo, contenido, reacciones) VALUES (1, 'Comencé a caminar', 'Hoy empecé la...	3 row(s) affected Records: 3 Duplicates: 0 Warnings: 0
2	12:49:24	SELECT * FROM foro LIMIT 0, 1000	3 row(s) returned

Ilustración 40 Tabla foro con registros exitosos

Tabla reporte:

```

332 VALUES
333 ('Pacientes', JSON_OBJECT('periodo', '2025-09'), '{"total":120,"activos":110}', 'JSON', 1),
334 ('Citas', JSON_OBJECT('periodo', '2025-09'), '{"programadas":40,"completadas":30}', 'JSON', 2),
335 ('Actividades', JSON_OBJECT('periodo', '2025-09'), '{"asignadas":60}', 'JSON', 3);
336
337 * SELECT * FROM reporte;
338
339

```

idReporte	tipo	parametros	contenido	formato	fechaGeneracion	idAdmin
1	Pacientes	['periodo': '2025-09']	['total':120,'activos':110]	JSON	2025-10-11 12:50:05	1
2	Citas	['periodo': '2025-09']	['programadas':40,'completadas':30]	JSON	2025-10-11 12:50:05	2
3	Actividades	['periodo': '2025-09']	['asignadas':60]	JSON	2025-10-11 12:50:05	3

Output

Action Output

#	Time	Action	Message
1	12:50:05	INSERT INTO reporte (tipo, parametros, contenido, formato, idAdmin) VALUES ('Pacientes',JSON_OBJECT(per...	3 row(s) affected Records: 3 Duplicates: 0 Warnings: 0
2	12:50:11	SELECT * FROM reporte	Error Code: 1064. You have an error in your SQL syntax; che
3	12:50:20	SELECT * FROM reporte LIMIT 0, 1000	3 row(s) returned

Ilustración 41 Tabla reporte con registros exitosos

Capítulo IV - Investigación de herramientas y tecnologías para el desarrollo del proyecto de TI.

En este capítulo se presenta la investigación de las herramientas y tecnologías que se consideraron para el desarrollo del proyecto HealthyView, una plataforma enfocada en el bienestar y la nutrición saludable. El propósito de esta sección es analizar distintas alternativas tecnológicas que permitan construir un sistema funcional, seguro y fácil de usar. A partir de esta investigación se seleccionaron las tecnologías más adecuadas para el desarrollo del proyecto, tomando en cuenta factores como su compatibilidad, facilidad de implementación, escalabilidad y costo.

4.1 Tecnologías investigadas

En esta sección se investigaron diversas herramientas y tecnologías que podrían aplicarse al desarrollo del proyecto HealthyView, con el propósito de comparar sus ventajas, compatibilidad y facilidad de implementación. Aunque no todas serán utilizadas, se analizaron para tener un panorama más amplio sobre las posibles soluciones tecnológicas que pueden emplearse en proyectos similares.

Lenguajes de programación

1) PHP

- **Nombre de la tecnología:** PHP (versión 8.3) – “Hypertext Preprocessor”.
- **Descripción general:** PHP es un lenguaje de programación interpretado, orientado principalmente al desarrollo de aplicaciones web del lado del servidor. Su propósito es generar contenido dinámico y comunicarse con bases de datos para crear sitios interactivos.
- **Historia y desarrollo:** Fue creado en 1994 por Rasmus Lerdorf, inicialmente como un conjunto de scripts para monitorear visitas a su sitio web. Con el tiempo evolucionó hasta convertirse en un lenguaje completo y de código abierto. Actualmente es mantenido por The PHP Group y sigue siendo uno de los lenguajes más utilizados en el desarrollo web, con soporte constante y amplia comunidad.

- **Aplicaciones:** PHP se utiliza para desarrollar sistemas de gestión de contenido (como WordPress o Joomla), aplicaciones web, tiendas en línea y plataformas de aprendizaje. En el proyecto HealthyView, PHP permitirá crear la lógica del servidor y gestionar la comunicación entre el usuario, la base de datos y las interfaces del sistema.

2) Python

- **Nombre de la tecnología:** Python (versión 3.12).

- **Descripción general:** Python es un lenguaje de programación interpretado, de alto nivel y multipropósito. Su sintaxis sencilla y legible facilita el aprendizaje y la creación de programas complejos en menos líneas de código. Se usa en ámbitos como desarrollo web, análisis de datos, automatización e inteligencia artificial.

- **Historia y desarrollo:** Fue creado por Guido van Rossum en 1991 con el objetivo de ofrecer un lenguaje fácil de usar y potente. A lo largo de los años ha sido mejorado por la comunidad global de desarrolladores y cuenta con gran cantidad de librerías que lo hacen muy versátil. Desde 2001 es gestionado por la Python Software Foundation.

- **Aplicaciones:** Python se emplea en áreas como ciencia de datos, inteligencia artificial, automatización de tareas, desarrollo web con frameworks como Django o Flask, y educación. Aunque no será el lenguaje principal en HealthyView, fue investigado por su potencial para futuras implementaciones relacionadas con el análisis de hábitos y estadísticas de salud.

3) JavaScript

- **Nombre de la tecnología:** JavaScript (versión ES2023).

- **Descripción general:** JavaScript es un lenguaje de programación interpretado que se ejecuta en el navegador del usuario, permitiendo agregar interactividad y dinamismo a las páginas web. Es parte esencial del desarrollo front-end y trabaja junto con HTML y CSS para mejorar la experiencia del usuario.

- **Historia y desarrollo:** Fue creado en 1995 por Brendan Eich mientras trabajaba en Netscape Communications. Con el paso del tiempo se convirtió en un estándar del desarrollo web, regulado por ECMA International bajo la especificación ECMAScript. Gracias a su evolución, hoy puede ejecutarse tanto en navegadores como en servidores mediante entornos como Node.js.

- **Aplicaciones:** JavaScript se usa en la creación de interfaces interactivas, validaciones de formularios, animaciones y comunicación asíncrona con el servidor (AJAX). También es clave en frameworks modernos como React, Vue o Angular. En el contexto de HealthyView, JavaScript podría emplearse para mejorar la interacción del usuario dentro del sitio web.

Entornos de Desarrollo

1) Visual Studio Code

- **Nombre de la tecnología:** Visual Studio Code (versión 1.93).

- **Descripción general:** Visual Studio Code es un editor de código fuente gratuito y multiplataforma desarrollado por Microsoft. Permite programar en distintos lenguajes como PHP, Python, JavaScript o HTML, y cuenta con extensiones que facilitan el desarrollo, depuración y control de versiones. Su propósito principal es ofrecer un entorno ligero pero potente para la creación de aplicaciones web y de escritorio.

- **Historia y desarrollo:** Fue lanzado en 2015 por Microsoft como un proyecto de código abierto basado en el framework Electron. Desde entonces ha ganado gran popularidad por su velocidad, personalización y compatibilidad con múltiples sistemas operativos. Su comunidad activa y su integración con Git y GitHub han permitido su evolución constante.

- **Aplicaciones:** Visual Studio Code se utiliza ampliamente para editar y depurar código en proyectos web, móviles y de software general. En el caso de HealthyView, se usa como entorno principal para escribir y organizar el código en PHP, HTML, CSS y JavaScript, facilitando el trabajo colaborativo y la productividad del equipo.

2) Sublime Text

- **Nombre de la tecnología:** Sublime Text (versión 4).
- **Descripción general:** Sublime Text es un editor de texto ligero y altamente personalizable que permite editar código en diversos lenguajes de programación. Ofrece una interfaz limpia, soporte para múltiples pestañas y herramientas que agilizan la escritura de código, como el autocompletado y la selección múltiple.
- **Historia y desarrollo:** Fue creado por Jon Skinner y lanzado en 2008. Desde su aparición, se ha caracterizado por su rapidez, estabilidad y capacidad de personalización mediante paquetes y plugins desarrollados por su comunidad. Aunque no es completamente gratuito, su versión de prueba es ampliamente utilizada.
- **Aplicaciones:** Sublime Text se emplea para editar y depurar código en proyectos pequeños o medianos. Es ideal para desarrolladores que buscan velocidad y simplicidad sin depender de entornos pesados. En HealthyView, fue investigado como una alternativa secundaria a Visual Studio Code por su eficiencia y portabilidad.

3) NetBeans

- **Nombre de la tecnología:** Apache NetBeans (versión 19).
- **Descripción general:** NetBeans es un entorno de desarrollo integrado (IDE) que proporciona herramientas completas para la creación de aplicaciones en distintos lenguajes, especialmente Java, PHP y HTML5. Incluye funciones como autocompletado, depuración, manejo de proyectos y diseño de interfaces gráficas.
- **Historia y desarrollo:** Originalmente desarrollado por estudiantes de la Universidad Charles en Praga en 1996, NetBeans fue adquirido por Sun Microsystems y más tarde por Oracle. Desde 2016 forma parte de la Apache Software Foundation, lo que garantizó su mantenimiento como software libre y su compatibilidad con nuevas tecnologías.

- **Aplicaciones:** NetBeans se utiliza en entornos académicos y profesionales para desarrollar aplicaciones empresariales y web. En el contexto de HealthyView, se analizó como alternativa por su integración con proyectos PHP y su soporte para bases de datos, aunque finalmente se optó por un editor más liviano y adaptable como Visual Studio Code.

Tecnologías de Presentación

1) HTML5

- **Nombre de la tecnología:** HTML5 (HyperText Markup Language, versión 5).

- **Descripción general:** HTML5 es el lenguaje estándar utilizado para definir la estructura y el contenido de las páginas web. Su función principal es organizar elementos como textos, imágenes, enlaces, formularios y multimedia dentro de un sitio. Además, permite integrar componentes interactivos y adaptativos sin necesidad de plugins externos.

- **Historia y desarrollo:** El lenguaje HTML fue creado a inicios de los años 90 por Tim Berners-Lee como base de la World Wide Web. La versión HTML5 comenzó su desarrollo en 2008 y fue publicada oficialmente por el World Wide Web Consortium (W3C) en 2014. Desde entonces, ha evolucionado para mejorar la compatibilidad, accesibilidad y soporte para dispositivos móviles.

- **Aplicaciones:** HTML5 se utiliza en prácticamente todos los sitios web del mundo. Permite la creación de estructuras sólidas y semánticas, lo que facilita el posicionamiento y la accesibilidad. En el proyecto HealthyView, HTML5 se usa para construir la estructura del sitio, como formularios de registro, secciones informativas y paneles de usuario.

2) CSS3

- **Nombre de la tecnología:** CSS3 (Cascading Style Sheets, versión 3).

- **Descripción general:** CSS3 es un lenguaje de hojas de estilo que define la apariencia y el diseño visual de las páginas web. Su propósito es separar el

contenido (HTML) de la presentación, permitiendo controlar colores, tipografías, márgenes, fondos, animaciones y la disposición de los elementos en pantalla.

- **Historia y desarrollo:** CSS fue introducido por el W3C en 1996 para complementar a HTML, ofreciendo control visual sobre las páginas. La versión CSS3, lanzada en 2011, incorporó nuevas propiedades como sombras, bordes redondeados, gradientes, transiciones y animaciones. Con el tiempo, se convirtió en una herramienta clave para el diseño moderno y adaptable a distintos dispositivos.

- **Aplicaciones:** CSS3 se emplea para crear interfaces atractivas, responsivas y coherentes con la identidad visual de un proyecto. En HealthyView, CSS3 se utiliza para dar estilo al sitio, aplicar paletas de colores suaves, distribuir los elementos y mejorar la experiencia visual del usuario.

3) Bootstrap

- **Nombre de la tecnología:** Bootstrap (versión 5.3).

- **Descripción general:** Bootstrap es un framework de diseño web que combina HTML, CSS y JavaScript para facilitar la creación de interfaces modernas y responsivas. Su principal función es ofrecer una colección de componentes y estilos prediseñados que permiten desarrollar sitios web más rápido y con un diseño profesional.

- **Historia y desarrollo:** Bootstrap fue creado en 2011 por Mark Otto y Jacob Thornton, desarrolladores de Twitter, con el propósito de unificar el diseño de sus herramientas internas. Posteriormente fue liberado como proyecto de código abierto y mantenido por la comunidad. Cada versión ha incorporado nuevas funciones adaptadas a las tendencias del diseño web.

- **Aplicaciones:** Bootstrap se utiliza ampliamente para crear sitios que se adapten automáticamente a distintos tamaños de pantalla (computadoras, tabletas y móviles). En HealthyView, fue investigado como una herramienta de apoyo para

agilizar el diseño de las páginas y mantener una presentación uniforme sin invertir demasiado tiempo en estilos manuales.

Base de Datos

1) MySQL

- **Nombre de la tecnología:** MySQL (versión 8.0).
- **Descripción general:** MySQL es un sistema de gestión de bases de datos relacional (RDBMS) basado en el lenguaje SQL. Su propósito principal es almacenar, organizar y administrar grandes volúmenes de información de manera estructurada. Se destaca por su velocidad, estabilidad y compatibilidad con distintos lenguajes de programación como PHP.
- **Historia y desarrollo:** MySQL fue creado en 1995 por Michael Widenius y David Axmark bajo la empresa MySQL AB. En 2008 fue adquirido por Sun Microsystems y posteriormente por Oracle Corporation. A lo largo del tiempo ha mantenido su carácter de software libre y es una de las bases de datos más utilizadas en el desarrollo web y empresarial.
- **Aplicaciones:** Se usa ampliamente en sitios web, sistemas de gestión de contenidos y plataformas en línea como WordPress, Joomla o Moodle. En el proyecto HealthyView, MySQL se empleará para almacenar la información de los usuarios, sus registros alimenticios, datos de progreso y configuración general del sistema.

2) PostgreSQL

- **Nombre de la tecnología:** PostgreSQL (versión 16).
- **Descripción general:** PostgreSQL es un sistema de gestión de bases de datos relacional de código abierto, reconocido por su alta confiabilidad, robustez y soporte para datos complejos. Su propósito es ofrecer una solución avanzada para manejar información estructurada y no estructurada, combinando estabilidad con características modernas como transacciones seguras y funciones personalizadas.

- **Historia y desarrollo:** Su origen se remonta al proyecto Ingres en la Universidad de California en Berkeley durante los años 80. En 1986 comenzó el desarrollo de Postgres, y más tarde evolucionó a PostgreSQL al incorporar soporte para el lenguaje SQL. Actualmente es mantenido por la comunidad global de desarrolladores y es considerado una alternativa poderosa frente a bases de datos comerciales.

- **Aplicaciones:** PostgreSQL se utiliza en aplicaciones empresariales, plataformas financieras, sistemas de análisis de datos y aplicaciones web complejas. En HealthyView, fue investigado por su capacidad para manejar información estructurada y escalar a futuro, aunque se optó por MySQL por su simplicidad y compatibilidad con el entorno XAMPP.

3) MongoDB

- **Nombre de la tecnología:** MongoDB (versión 7.0).

- **Descripción general:** MongoDB es un sistema de gestión de bases de datos NoSQL orientado a documentos. A diferencia de los sistemas relacionales, almacena la información en formato JSON, lo que permite mayor flexibilidad al manejar datos no estructurados o cambiantes. Está diseñado para ofrecer alta disponibilidad y rendimiento en aplicaciones con grandes volúmenes de datos.

- **Historia y desarrollo:** MongoDB fue desarrollado en 2007 por la empresa 10gen, hoy conocida como MongoDB Inc.. Desde su lanzamiento, se convirtió en una de las bases NoSQL más populares gracias a su facilidad de uso, escalabilidad y comunidad activa. Con los años ha incorporado características como replicación, balanceo de carga y agregaciones avanzadas.

- **Aplicaciones:** MongoDB se utiliza en sistemas que requieren manejar datos flexibles o en tiempo real, como aplicaciones móviles, análisis de comportamiento de usuarios o gestión de contenido dinámico. En el caso de HealthyView, fue investigado como una alternativa moderna para futuras versiones que integren recomendaciones personalizadas o seguimiento de hábitos más detallado.

Servidor y Herramientas de Integración

1) XAMPP

- **Nombre de la tecnología:** XAMPP (versión 8.2).
- **Descripción general:** XAMPP es un paquete de software libre que incluye las principales herramientas necesarias para desarrollar y ejecutar aplicaciones web de forma local. Integra el servidor Apache, el lenguaje PHP, el sistema gestor de bases de datos MySQL (o MariaDB) y otros componentes como phpMyAdmin y FileZilla. Su objetivo es simplificar la instalación y configuración de un entorno de desarrollo completo.
- **Historia y desarrollo:** Fue creado por el grupo Apache Friends en 2002 como una alternativa sencilla para configurar servidores locales sin necesidad de conocimientos avanzados. Su nombre proviene de las iniciales de los componentes que lo integran: X (multiplataforma), A (Apache), M (MySQL), P (PHP) y P (Perl). Con el tiempo se ha convertido en una herramienta esencial para desarrolladores web.
- **Aplicaciones:** XAMPP se utiliza para crear y probar aplicaciones web en un entorno local antes de subirlas a un servidor real. Permite ejecutar sitios desarrollados con PHP y bases de datos MySQL sin conexión a Internet. En HealthyView, se usa como entorno principal de desarrollo y pruebas del sistema, asegurando que todas las funciones trabajen correctamente antes de su implementación final.

2) WAMP

- **Nombre de la tecnología:** WAMP (versión 3.3).
- **Descripción general:** WAMP es un conjunto de software que proporciona un entorno de desarrollo web para sistemas operativos Windows. Incluye Apache, MySQL y PHP, además de herramientas gráficas para gestionar bases de datos y servicios del servidor. Su propósito es permitir a los desarrolladores crear sitios web dinámicos sin necesidad de configuraciones complejas.

- **Historia y desarrollo:** El término WAMP surgió como adaptación de LAMP (Linux, Apache, MySQL y PHP) para funcionar en Windows. Desde su creación, ha sido una alternativa ligera y práctica para quienes prefieren trabajar en este sistema operativo. Con el tiempo, ha integrado mejoras en seguridad y facilidad de uso para desarrolladores principiantes.

- **Aplicaciones:** WAMP se utiliza principalmente para el desarrollo y prueba de aplicaciones web en entornos Windows. Es ideal para proyectos PHP con MySQL y permite administrar el servidor Apache desde una interfaz gráfica. En HealthyView, fue investigado como una alternativa a XAMPP, pero se eligió este último por su compatibilidad multiplataforma y facilidad de configuración.

3) Laragon

- **Nombre de la tecnología:** Laragon (versión 6.0).

- **Descripción general:** Laragon es un entorno de desarrollo web portátil y moderno que permite ejecutar aplicaciones basadas en PHP, MySQL, Apache y otras tecnologías. Se caracteriza por su rapidez, estabilidad y facilidad de instalación, ofreciendo un servidor local que se inicia automáticamente y funciona sin requerir configuraciones manuales.

- **Historia y desarrollo:** Laragon fue desarrollado por Leo Khoa y lanzado en 2015 como una alternativa más ligera y eficiente a XAMPP y WAMP. Su diseño modular permite integrar fácilmente nuevos componentes, como Node.js, Nginx o PostgreSQL. Gracias a su enfoque en el rendimiento, ha ganado popularidad entre los desarrolladores web modernos.

- **Aplicaciones:** Laragon se utiliza para ejecutar y probar aplicaciones web en un entorno local optimizado. Es especialmente útil en proyectos que requieren varias versiones de PHP o diferentes servidores web. En HealthyView, fue investigado como una herramienta de desarrollo alternativa por su ligereza y capacidad de personalización, aunque no se seleccionó como entorno principal.

Servidor Web

1) Apache

- **Nombre de la tecnología:** Apache HTTP Server (versión 2.4).
- **Descripción general:** Apache es un servidor web de código abierto que permite alojar, gestionar y distribuir páginas web a través del protocolo HTTP. Es uno de los servidores más utilizados en el mundo gracias a su estabilidad, seguridad y compatibilidad con diferentes lenguajes, como PHP, Python y Perl.
- **Historia y desarrollo:** El servidor Apache fue creado en 1995 por el grupo Apache Software Foundation, con el objetivo de ofrecer una alternativa gratuita y confiable para el alojamiento web. Con el paso de los años ha evolucionado, integrando módulos que amplían sus capacidades, como la gestión de seguridad, redirección, compresión y soporte para HTTPS.
- **Aplicaciones:** Apache se utiliza para ejecutar sitios web, aplicaciones en línea y sistemas internos. Permite la configuración de dominios, control de accesos y monitoreo de tráfico. En el proyecto HealthyView, Apache forma parte del entorno XAMPP, funcionando como el servidor principal encargado de ejecutar el código PHP y mostrar el contenido al usuario.

2) Nginx

- **Nombre de la tecnología:** Nginx (versión 1.26).
- **Descripción general:** Nginx es un servidor web y proxy inverso de alto rendimiento, diseñado para manejar grandes cantidades de conexiones simultáneas con bajo consumo de recursos. Su arquitectura basada en eventos lo hace ideal para sitios que requieren velocidad, equilibrio de carga y alta disponibilidad.
- **Historia y desarrollo:** Fue desarrollado por Igor Sysoev en 2004 con el objetivo de resolver el problema del rendimiento en servidores tradicionales. Desde su lanzamiento, Nginx ha ganado popularidad en proyectos de gran escala y

actualmente es mantenido por la empresa F5 Networks y una comunidad activa de desarrolladores.

- **Aplicaciones:** Nginx se utiliza en servidores de alto tráfico, plataformas de streaming y aplicaciones web complejas que requieren gran capacidad de respuesta. En el contexto de HealthyView, fue investigado como una opción moderna y eficiente, aunque se prefirió Apache por su integración directa con XAMPP y su facilidad de uso en entornos educativos.

3) IIS (Internet Information Services)

- **Nombre de la tecnología:** Internet Information Services (IIS) – versión 10.0.

- **Descripción general:** IIS es un servidor web desarrollado por Microsoft para el sistema operativo Windows. Permite alojar sitios web, aplicaciones y servicios basados en tecnologías como ASP.NET, PHP y HTML. Está diseñado para ofrecer un entorno seguro y bien integrado con otras herramientas del ecosistema Microsoft.

- **Historia y desarrollo:** IIS fue lanzado por primera vez en 1995 como parte de Windows NT 3.51. Con el tiempo, se ha actualizado para incluir soporte para nuevos protocolos, administración remota y mejores mecanismos de seguridad. Hoy en día, sigue siendo una opción confiable en entornos empresariales que utilizan infraestructura de Microsoft.

- **Aplicaciones:** IIS se utiliza principalmente en empresas o instituciones que trabajan con servidores Windows y tecnologías .NET. Permite administrar sitios web, realizar configuraciones personalizadas y controlar permisos de acceso. En HealthyView, fue investigado como una alternativa para entornos Windows, aunque no se seleccionó por no ser multiplataforma y requerir licencias específicas.

4.2 Selección de tecnologías

En esta sección se presentan las tecnologías seleccionadas para el desarrollo del proyecto HealthyView, elegidas con base en los resultados del análisis realizado en la fase de investigación. La selección se fundamenta en criterios como la compatibilidad entre herramientas, facilidad de uso, disponibilidad, costo, estabilidad y adecuación a los objetivos del sistema.

Cada tecnología fue elegida por su capacidad para aportar funcionalidad, eficiencia y fiabilidad al proyecto, garantizando un entorno de desarrollo óptimo y coherente con las necesidades técnicas del sistema web. A continuación, se describen las tecnologías seleccionadas, junto con una justificación detallada que explica por qué cada una resulta la más apropiada para la construcción y correcto funcionamiento de HealthyView.

Lenguaje de programación

- **Nombre:** PHP (versión 8.3)
- **Justificación:** Se seleccionó PHP porque es un lenguaje especializado en el desarrollo de aplicaciones web dinámicas y se integra de manera nativa con MySQL y el servidor Apache, herramientas también empleadas en el proyecto. Además, es de código abierto, ampliamente documentado y compatible con múltiples plataformas, lo que facilita su aprendizaje y mantenimiento. Comparado con otros lenguajes como Python o JavaScript, PHP ofrece una curva de aprendizaje más sencilla, menor consumo de recursos en ejecución y un entorno estable dentro de XAMPP, permitiendo construir la lógica del sistema HealthyView de forma práctica y eficiente.

Entorno de desarrollo

- **Nombre:** Visual Studio Code (versión 1.93)
- **Justificación:** Se eligió Visual Studio Code por ser un editor de código moderno, ligero y gratuito que admite extensiones para PHP, HTML, CSS y JavaScript, los

principales lenguajes del proyecto. Su interfaz personalizable y su integración con Git permiten un flujo de trabajo ágil y organizado. A diferencia de entornos más pesados como NetBeans o Eclipse, VS Code brinda un rendimiento más fluido, arranque rápido y una experiencia optimizada para estudiantes y equipos pequeños, siendo ideal para el desarrollo de HealthyView.

Lenguajes de presentación

- **Nombre:** HTML5 (versión 5)

- **Justificación:** HTML5 se seleccionó como el lenguaje base para la estructura del sistema web debido a su compatibilidad universal y su capacidad para trabajar junto con CSS y PHP. Permite diseñar interfaces accesibles, semánticas y adaptables a distintos dispositivos. A diferencia de versiones anteriores o tecnologías propietarias, HTML5 ofrece soporte nativo para formularios, video, audio y etiquetas modernas que mejoran la experiencia del usuario, elementos clave para el diseño limpio y funcional de HealthyView.

- **Nombre:** CSS3 (versión 3)

- **Justificación:** CSS3 fue elegido por su potencia para definir la apariencia visual del sitio, mediante estilos modernos, animaciones y estructuras responsivas. Su integración directa con HTML5 permite mantener la presentación separada del contenido, facilitando la personalización del diseño. Frente a frameworks visuales como Bootstrap, se optó por CSS3 puro para tener un control más directo sobre los estilos y mantener el sitio ligero, garantizando una identidad visual coherente con los objetivos estéticos y de accesibilidad de HealthyView.

Base de datos

- **Nombre:** MySQL (versión 8.0)

- **Justificación:** Se seleccionó MySQL como sistema gestor de base de datos por su estabilidad, seguridad y compatibilidad directa con PHP y XAMPP. Permite crear estructuras relacionales ideales para almacenar información de usuarios, registros

alimenticios, sesiones y configuraciones del sistema. En comparación con PostgreSQL o MongoDB, MySQL destaca por su facilidad de uso, su amplia documentación y el soporte nativo dentro del entorno de desarrollo local. Estas características lo convierten en la opción más adecuada para el tamaño y alcance del proyecto *HealthyView*.

Servidor local

- **Nombre:** XAMPP (versión 8.2, incluye Apache 2.4)
- **Justificación:** XAMPP se eligió porque integra en un solo paquete todo lo necesario para ejecutar el proyecto localmente: Apache, PHP y MySQL. Su instalación sencilla y compatibilidad multiplataforma permiten probar y depurar el sistema sin requerir configuraciones complejas. Frente a alternativas como WAMP o Laragon, XAMPP es la opción más estable y utilizada en entornos académicos, ideal para estudiantes y equipos pequeños. Su facilidad de administración y su entorno visual intuitivo hacen posible desarrollar, probar y desplegar HealthyView de forma eficiente.

Servidor web

- **Nombre:** Apache (versión 2.4, integrado en XAMPP)
- **Justificación:** Se seleccionó Apache como servidor web por su fiabilidad, seguridad y compatibilidad directa con PHP. Es uno de los servidores más usados a nivel mundial y está completamente integrado dentro de XAMPP, lo que simplifica su configuración y ejecución local. Comparado con otros servidores como Nginx o IIS, Apache destaca por su estabilidad, facilidad de administración y amplia documentación. En HealthyView, Apache cumple el rol de interpretar las peticiones del cliente, procesar los archivos PHP y enviar las respuestas al navegador de manera rápida y segura.

Conclusión

Como conclusión se logró integrar de manera coherente los conocimientos adquiridos para diseñar una propuesta tecnológica útil y funcional. A través de la investigación y selección de herramientas adecuadas, se eligieron tecnologías que permiten desarrollar una plataforma confiable, escalable y fácil de usar, enfocada en mejorar los hábitos y el bienestar de los usuarios.

El análisis realizado ayudó a comprender la importancia de elegir soluciones tecnológicas que realmente respondan a las necesidades del proyecto y a las problemáticas detectadas, como la falta de control en la alimentación o la dificultad para mantener una rutina saludable. Estas decisiones no solo fortalecen el desempeño del sistema, sino que también garantizan su crecimiento y adaptación a futuro.

Finalmente, este proceso reafirmó el valor de mantenerse en constante aprendizaje dentro del ámbito de las Tecnologías de la Información. La capacidad de investigar, comparar y aplicar nuevas herramientas demuestra que, más allá del desarrollo de un sistema, lo esencial es seguir creciendo como profesionales capaces de adaptarse a los cambios y aprovechar las oportunidades que ofrece la innovación tecnológica.

Bibliografía

- Upemor :: Inicio. (s. f.). <https://www.upemor.edu.mx/>

- Gil-Romo, J. (2007). Estudios sobre alimentación y nutrición en México: una mirada a través del género. *Salud Pública de México*, 49(6), 445-453.
https://www.scielo.org.mx/scielo.php?script=sci_arttext&pid=S0036-36342007000600012

- López-Gutiérrez, J., Castañeda-Limón, D., López-Aguilar, M., & López-García, J. (2017). Conocimientos nutricionales en estudiantes universitarios del sector público del Estado de Chiapas, México. *Investigación en Educación Médica*, 6(22), 118-125.
<https://www.studocu.com/es-mx/document/universidad-latina/mercadotecnia/4-como-redactar-los-antecedentes-de-la-investigacion/25921021>

- OpenAI. (2025). *ChatGPT (GPT-5) [Modelo de lenguaje grande]*. OpenAI.
<https://chat.openai.com/>

- Apache Software Foundation. (2024). *Apache HTTP Server Documentation (Version 2.4)*.
<https://httpd.apache.org/docs/>

- Apache Friends. (2024). *XAMPP: A free and open-source cross-platform web server solution stack package*. <https://www.apachefriends.org/>

- MySQL AB / Oracle Corporation. (2024). *MySQL Reference Manual (Version 8.0)*.
<https://dev.mysql.com/doc/>

- PHP Group. (2024). *PHP Manual (Version 8.3)*. <https://www.php.net/manual/en/>

- Microsoft. (2024). *Visual Studio Code Documentation (Version 1.93)*.

<https://code.visualstudio.com/docs>

- World Wide Web Consortium (W3C). (2024). *HTML5 Specification*.

<https://www.w3.org/TR/html52/>

- Python Software Foundation. (2024). *Python 3.12 Documentation*.

<https://docs.python.org/3/>